

ON EFFECTIVELY COMPUTING THE ANALYTIC CENTER OF THE SOLUTION SET BY PRIMAL-DUAL INTERIOR-POINT METHODS*

MARÍA D. GONZÁLEZ-LIMA[†], RICHARD A. TAPIA[‡], AND FLORIAN A. POTRA[§]

Abstract. The computation of the analytic center of the solution set can be important in linear programming applications where it is desirable to obtain a solution that is not near the relative boundary of the solution set. In this work we discuss the effective computation of the analytic center solution by the use of primal-dual interior-point methods. A primal-dual interior-point algorithm designed for effectively computing the analytic-center solution is proposed, and theory and numerical results are presented.

Key words. primal-dual interior-point algorithm, analytic-center solution, linear programming

AMS subject classifications. 65K, 49M, 90C

PII. S1052623495291793

1. Introduction. The computation of the analytic center of the solution set is valuable for some linear programming applications where it is desirable to obtain a solution that is not near the relative boundary of the solution set (see Charnes, Cooper, and Thrall [3] and Thompson, Dharmapala, and Thrall [19] for such an application). In this work we discuss the use of primal-dual interior-point methods for effectively computing the analytic-center solution.

Because in this work we are interested in using primal-dual interior-point methods for finding a specific solution (the analytic-center solution), the study of the convergence of the iteration sequence generated by primal-dual interior-point methods needs to be addressed. It is relevant to mention that this is not the case for many real-world problems where any feasible point with the value of the duality gap close to zero is a satisfactory approximate solution.

The basic primal-dual interior-point method for linear programming was proposed by Kojima, Mizuno, and Yoshise [11]. Most primal-dual interior-point methods fall into the general framework of the Kojima–Mizuno–Yoshise algorithm. The first work devoted to the study of the convergence of the iteration sequence generated by the Kojima–Mizuno–Yoshise algorithm is Tapia, Zhang, and Ye [18]. Later on, Zhang and Tapia [24] derived conditions under which this iteration sequence converged to the analytic-center solution, assuming that the sequence converged. However, the conditions given in [24] are not compatible with the Tapia–Zhang–Ye conditions for the convergence of the iteration sequence. The former requires the centering parameters

*Received by the editors September 8, 1995; accepted for publication (in revised form) October 29, 1996. This research was partially supported by NSF Cooperative agreement no. CCR-88-09615, NSF grant DMS 9305760, ARO grant 9DAAL03-90-G-0093, DOE grant DEFG05-86-ER25017, and AFOSR grant 89-0363.

<http://www.siam.org/journals/siopt/8-1/29179.html>

[†]Departamento de Cómputo Científico y Estadística y Centro de Estadística y Software Matemático (CESMA), Universidad Simón Bolívar, Apdo 89000, Caracas 1080-A, Venezuela (mg1@cesma.usb.ve). This author was partially supported by Fulbright/LASPAU.

[‡]Department of Computational and Applied Mathematics and Center for Research on Parallel Computation, Rice University, Houston, TX 77251-1892 (rat@rice.edu).

[§]Department of Mathematics and Department of Computer Science, The University of Iowa, Iowa City, IA 52242 (potra@math.uiowa.edu).

to be bounded away from zero, while the latter requires that these same parameters converge to zero.

An interesting variant of the Kojima–Mizuno–Yoshise algorithm was proposed by Mizuno, Todd, and Ye [16]. The Mizuno–Todd–Ye predictor-corrector algorithm is a path-following algorithm which keeps the iterates in a small neighborhood of the central path. Zhang and El-Bakry [22] showed that a modified version of the Mizuno–Todd–Ye predictor-corrector algorithm had the property that the iteration sequence that it generated converged to the analytic center. Soon after the result of Zhang and El-Bakry, Gonzaga and Tapia [8] proved that the Mizuno–Todd–Ye predictor-corrector algorithm itself, without any modification, generated an iteration sequence which converged to the analytic center. These are the first results proving convergence of the iteration sequence to the analytic-center solution for any primal-dual interior-point method.

The Mizuno–Todd–Ye predictor-corrector algorithm has two characteristics that make it unappealing for most practical purposes. The first one is that at each iteration the algorithm requires the solution of two linear systems of equations where the matrices are of the same form as those for the Kojima–Mizuno–Yoshise algorithm. Hence, the cost per iteration is twice that of the Kojima–Mizuno–Yoshise algorithm. The second deficiency is the requirement that the starting feasible point must be inside a specified β -neighborhood of the central path.

Recently, Bonnans and Gonzaga [1] gave sufficient conditions for convergence and convergence to the analytic center of the iteration sequence generated by primal-dual interior-point algorithms from a general class of algorithms containing the Kojima–Mizuno–Yoshise and Mizuno–Todd–Ye algorithms. Their result is an extension of the work of Gonzaga and Tapia [8]. We have found that theoretical results are often limited in practice. Theorem 3.2 in Bonnans and Gonzaga [1] guarantees the theoretical convergence of the iteration sequence to the analytic center assuming, among other conditions, that the iterates stay in a small neighborhood of the central path. However, strong adherence of the iterates to the central path results in slow performance of the algorithm, especially when this adherence is also maintained far from the solution set. In addition, we will show that although the iteration sequence generated by the Kojima–Mizuno–Yoshise algorithm may theoretically converge to the analytic center, in practice this convergence is not, for many cases, easy to attain. This negative behavior is related to the proximity of the iterates to the solution set. If the sequence generated by the algorithm approaches the solution set very fast and is far from the analytic center, the stopping criteria are met before the approximate solution has a chance of getting close to the analytic center. Also, the singularity of the Jacobian matrix at the solution may cause serious numerical problems that affect the computation of the analytic center when the iterates are very close to the solution set.

Hence, a balance between proximity to the central path and proximity to the solution set is crucial for an effective computation of the analytic center of the solution set using a primal-dual interior-point algorithm.

In this work a modification of the Kojima–Mizuno–Yoshise primal-dual algorithm is proposed. This modified algorithm combines the objectives of approaching the central path and the solution set to effectively compute the analytic-center solution. We prove that the proposed algorithm generates an iteration sequence with the property that a subsequence converges to the analytic center of the solution set; however, in practice we always have that the entire sequence converges to the analytic cen-

ter. Moreover, we also prove that with a small modification the algorithm achieves polynomial complexity. Finally, we demonstrate that the numerical behavior of the algorithm is quite good.

This paper is organized as follows. The next section contains some preliminary background and notation. In section 3 our proposed primal-dual interior-point algorithm designed for effectively computing the analytic center is described. The convergence properties for the algorithm are studied in section 4. In section 5 we prove that the algorithm has polynomial complexity if the algorithm is modified in an appropriate way. In section 6 we study the numerical behavior of the proposed algorithm. Also, we compare the numerical behavior of the proposed algorithm with the numerical behavior of the Kojima–Mizuno–Yoshise and the Mizuno–Todd–Ye algorithms. Finally, some conclusions and remarks are given in section 7.

2. Preliminaries and notation. We consider the linear programming problem in the standard form

$$(2.1) \quad \begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b, \quad x \geq 0, \end{array}$$

where $c, x \in \mathbf{R}^n$, $b \in \mathbf{R}^m$, $A \in \mathbf{R}^{m \times n}$ ($m < n$).

The Karush–Kuhn–Tucker optimality conditions for (2.1) are

$$(2.2) \quad F(x, y, z) = \begin{pmatrix} Ax - b \\ A^T y + z - c \\ XZe \end{pmatrix} = 0, \quad (x, z) \geq 0,$$

where $X = \text{diag}(x)$, $Z = \text{diag}(z)$, and $e = (1, 1, \dots, 1)^T \in \mathbf{R}^n$.

The feasibility set of problem (2.2) is

$$\mathcal{F} = \{(x, y, z) : Ax = b, A^T y + z = c, (x, z) \geq 0\}.$$

A feasible point $(x, y, z) \in \mathcal{F}$ is said to be strictly feasible if x and z are strictly positive. The set of all the strictly feasible points is denoted by $\mathcal{F}^+$.

We denote the solution set of problem (2.2) by

$$\mathcal{S} = \{(x, y, z) : F(x, y, z) = 0, (x, z) \geq 0\}.$$

A point (x, y, z) in $\mathcal{S}$ is said to be a strict complementarity solution if $x_i + z_i > 0$ for all $i = 1, \dots, n$.

The following standard conditions are assumed:

- (A1) $\{x \in \mathbf{R}^n : Ax = b, x > 0\} \neq \emptyset$, and
- (A2) $\{(y, z) \in \mathbf{R}^{m+n} : A^T y + z = c, z > 0\} \neq \emptyset$.
- (A3) A has full rank m .

We are particularly concerned with the case when $\mathcal{S}$ is not a singleton set.

Under the previous assumptions, the solution set $\mathcal{S}$ has the following interesting structure:

- (i) The set $\mathcal{S} \neq \emptyset$ is bounded.
- (ii) All points in the relative interior of $\mathcal{S}$ are strict complementarity solutions and all points on the relative boundary of $\mathcal{S}$ are not.
- (iii) The zero–nonzero pattern of points in the relative interior of $\mathcal{S}$ is invariant.

Proof. See El-Bakry, Tapia, and Zhang [6] for proofs. $\square$

Therefore, for any $(x^*, y^*, z^*) \in ri(\mathcal{S})$, where $ri(\mathcal{S})$ denotes the relative interior of $\mathcal{S}$, the following index sets

$$I_x^+ = \{i : x_i^* > 0, 1 \leq i \leq n\} \quad \text{and} \quad I_z^+ = \{i : z_i^* > 0, 1 \leq i \leq n\}$$

are independent of the choice of (x^*, y^*, z^*) . Moreover, by strict complementarity

$$I_x^+ \cup I_z^+ = \{1, 2, \dots, n\} \quad \text{and} \quad I_x^+ \cap I_z^+ = \emptyset.$$

Because of this structure of the solution set $\mathcal{S}$, the solutions in $ri(\mathcal{S})$ may be characterized as having a maximal number of nonzero components. Among them, there is a solution that may be thought of as the center-most solution in the sense that it maximizes the product of the positive components. This solution is called the analytic center of the solution set and was studied by McLinden [13] in a general setting and later independently by Sonnevend [17] in the context of linear programming. Formally, the analytic center of the solution set $\mathcal{S}$ is defined as

$$(2.3) \quad (x^*, y^*, z^*) = \arg \max \{ \psi(x, z) : (x, y, z) \in \mathcal{S} \},$$

where

$$\psi(x, z) = \prod_{i \in I_x^+} x_i \prod_{i \in I_z^+} z_i.$$

Equivalently, in (2.3) one can replace $\psi(x, z)$ by its logarithm; i.e.,

$$\ln \psi(x, z) = \sum_{i \in I_x^+} \ln x_i + \sum_{i \in I_z^+} \ln z_i.$$

The central path associated with problem (2.2) parameterized by the parameter μ is defined as the collection of points $\mathcal{P}_c = \{\mathcal{P}_c(\mu); \mu > 0\}$, where

$$\mathcal{P}_c(\mu) = (x(\mu), y(\mu), z(\mu)) \text{ is a strictly feasible point satisfying } X(\mu)Z(\mu)e = \mu e.$$

This is equivalent to saying that a strictly feasible point (x, y, z) is on the central path (for some $\mu > 0$) if and only if it satisfies $x_1 z_1 = x_2 z_2 = \dots = x_n z_n$.

These notions of analytic center and central path are well defined under assumptions (A1) and (A2). For more details see McLinden [13], Megiddo [15], and Gonzaga [7].

A β -neighborhood of the central path is defined as

$$(2.4) \quad \mathcal{N}(\beta) = \{(x, y, z) : (x, y, z) \in \mathcal{F}, \quad \|Xz - \mu e\|_2 \leq \beta \mu\},$$

where $\beta \in (0, 1)$ and $\mu = x^T z / n$.

A very interesting result is Theorem 9 of McLinden [13]. In the case of linear programming, it states that the central path intersects the solution set at the analytic center; i.e., the central-path point $(x(\mu), y(\mu), z(\mu))$ converges to the analytic center (x^*, y^*, z^*) as μ converges to zero. See also Proposition 8.2 in Megiddo [15] and the discussion preceding it. This fact plays a critical role in the development of most primal-dual interior-point algorithms which attempt to follow the central path.

3. Algorithm. The interior-point algorithm proposed in this work has been designed for effectively computing the analytic center of the solution set in linear programming and it is a modification of the Kojima–Mizuno–Yoshise algorithm.

The following lemma can be found in Zhang and Tapia [24]. It provides a sufficient condition for a strictly feasible sequence $\{(x^k, y^k, z^k)\}$ to converge to the analytic center of the solution set.

LEMMA 3.1 (Zhang–Tapia). *Let $\{(x^k, y^k, z^k)\}$ be a sequence of strictly feasible points. Let*

$$(3.1) \quad \mu^k = (x^k)^T z^k / n \rightarrow 0$$

and

$$(3.2) \quad \|X^k z^k / \mu^k - e\|_2 \rightarrow 0$$

with $e = (1, \dots, 1)^T \in \mathbf{R}^n$. Then $\{(x^k, y^k, z^k)\}$ converges to the analytic center of the solution set.

Our approach to constructing an algorithm for computing the analytic center is to attempt to enforce conditions (3.1) and (3.2).

A key ingredient in the proposed algorithm is the use of the damped Newton method applied to the perturbed Karush–Kuhn–Tucker conditions

$$F_\mu(x, y, z) = \begin{pmatrix} Ax - b \\ A^T y + z - c \\ XZe - \mu e \end{pmatrix} = 0$$

with $\mu > 0$. A line-search technique is used as a globalization of Newton's method. For fixed $\mu > 0$, the merit function considered is $f_\mu : \mathbf{R}^{2n+m} \rightarrow \mathbf{R}$ defined by

$$f_\mu(x, y, z) = \|F_\mu(x, y, z) / \mu\|_2^2.$$

Observe that

$$(3.3) \quad f_\mu(x, y, z) = \|(Xz - \mu e) / \mu\|_2^2 \quad \text{for } (x, y, z) \in \mathcal{F}.$$

The function f_μ measures proximity or closeness to the central path at μ .

The basic idea of the algorithm is to consider a sequence of gradually shrinking neighborhoods $\{\mathcal{N}(\beta^j)\}$ of the central path with $\beta^j \rightarrow 0$. In the next section we will show that under certain assumptions the proposed algorithm generates an iteration sequence $\{(x^k, y^k, z^k)\}$ such that a subsequence $(x^{k_j}, y^{k_j}, z^{k_j})$ belongs to $\mathcal{N}(\beta^j)$. This subsequence is obtained by considering a sequence of μ 's and for each fixed μ using a damped Newton's method (with a line-search globalization strategy) to approximately solve the system $\mathcal{F}_\mu(x, y, z) = 0$.

A description of the algorithm is the following.

ALGORITHM 3.1 (long-step shrinking-neighborhood (LSSN) algorithm). *Consider a strictly feasible point $w^0 = (x^0, y^0, z^0)$, $\sigma^0 \in (0, 1)$, and $\beta^0 \in (0, 1)$.*

Choose $\eta \in (0, 1/2)$ and $0 < l < u < 1$.

Do until convergence

(0) *Set $k = 0$.*

(1) *Define parameters: Set $\mu = \mu^k = \sigma^0 (x^k)^T z^k / n$ and $\beta = \beta^k$.*

(2) *Check proximity to the central path: If*

$$(3.4) \quad \|X^k Z^k e / \mu - e\|_2 \leq \beta,$$

go to (6).

(3) *Compute new iterate:*

(3.a) Solve the following system for $\Delta w^k = (\Delta x^k, \Delta y^k, \Delta z^k)$:

$$F'(x^k, y^k, z^k) \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = -F(x^k, y^k, z^k) + \mu \begin{pmatrix} 0 \\ 0 \\ e \end{pmatrix}.$$

(3.b) Choose $\tau^k \in (0, 1)$ and compute the step-length $\alpha^k = \min(1, \tau^k \hat{\alpha}^k)$, where

$$\hat{\alpha}^k = \frac{-1}{\min((X^k)^{-1} \Delta x^k, (Z^k)^{-1} \Delta z^k)}.$$

(3.c) Form the iterate

$$w^{k+1} = (x^{k+1}, y^{k+1}, z^{k+1}) = (x^k, y^k, z^k) + \alpha^k (\Delta x^k, \Delta y^k, \Delta z^k).$$

(4) Line search:

(4.a) If

$$(3.5) \quad f_\mu(w^{k+1}) \leq f_\mu(w^k) + \eta \alpha^k \nabla f_\mu(w^k)^T (\Delta x^k, \Delta y^k, \Delta z^k)^T,$$

go to (5).

(4.b) If not, reduce $\alpha^k := \rho \alpha^k$ with $\rho \in [l, u]$ and form the iterate

$$w^{k+1} = (x^{k+1}, y^{k+1}, z^{k+1}) = (x^k, y^k, z^k) + \alpha^k (\Delta x^k, \Delta y^k, \Delta z^k).$$

(4.c) Go to step (4.1).

(5) Set $k := k + 1$, $\beta^k = \beta$, $\mu^k = \mu$, and go to (2).

(6) Set $\mu = \mu^k = \sigma^0(x^k)^T z^k / n$.

(7) Do (3.1), (3.2), (3.3).

(8) Decrease β -neighborhood: Choose $\beta^{k+1} \leq \beta^k$.

(9) Set $k := k + 1$, and go to (1).

We note that while condition (3.4) is not satisfied, substeps (3) and (4) of the above algorithm are performed iteratively with the same value of μ . It is convenient to denote by $\mathcal{I}(\mu)$ the corresponding set of indices; i.e.,

$$(3.6) \quad \mathcal{I}(\mu) = \{k : \mu^k = \mu\}.$$

In the next section we will show that the index set $\mathcal{I}(\mu)$ is finite for each μ chosen by Algorithm 3.1. This means that under certain assumptions the iteration sequence generated by the LSSN algorithm, when $\mu > 0$ is fixed, approaches the central path in the sense that condition (3.4) (or proximity to the central path) is satisfied.

4. Convergence properties. In this section we prove that under certain assumptions the LSSN algorithm generates an iteration sequence which contains a subsequence that converges to the analytic center of the solution set. The convergence result is established by showing that under certain assumptions there exists a subsequence satisfying conditions (3.1) and (3.2) of Lemma 3.1. Therefore, we need to study the behavior of the proposed algorithm with respect to the two objectives of decreasing the gap and centering the iterates. In the following two propositions we study these issues. In the first proposition we prove that the gap sequence generated by the LSSN algorithm is a decreasing sequence. In the second proposition we prove that for each fixed $\mu > 0$, the search direction is a descent direction for the function

f_μ and the sequence $\{f_\mu(x^k, y^k, z^k)\}$ generated by the algorithm is a strictly decreasing sequence. Also, we show that $f_\mu(x^k, y^k, z^k)$ is an upper bound for the Euclidean measure $\|\frac{X^k z^k}{(x^k)^T z^k / n} - e\|_2^2$; therefore, condition (3.2) may be established for any sequence of iterates $\{(x^k, y^k, z^k)\}$ with the property that the corresponding sequence $\{f_\mu(x^k, y^k, z^k)\}$ converges to zero.

PROPOSITION 4.1. *Let $\{(x^k, y^k, z^k)\}$ and $\{\alpha^k\}$ be generated by the LSSN algorithm (Algorithm 3.1). Then,*

$$(x^{k+1})^T z^{k+1} = (1 - \alpha^k(1 - \sigma^k))(x^k)^T z^k \text{ for all } k \geq 0 \text{ and for some } \sigma^k \in (0, 1].$$

Proof. Define the following sets:

$$I_{cp} = \{k \geq 0 : (x^k, y^k, z^k) \text{ satisfies condition (3.4) in the algorithm}\}; \text{ i.e.,}$$

$$I_{cp} = \{k \geq 0 : \|X^k Z^k e / \mu - e\|_2 \leq \beta^k\}, \text{ and}$$

$$I_{cp}^+ = \{k \geq 1 : (x^{k-1}, y^{k-1}, z^{k-1}) \text{ satisfies condition (3.4) in the algorithm}\} \cup \{0\}.$$

For each k we denote by k_{cp}^+ the largest element in I_{cp}^+ such that $k_{cp}^+ \leq k$ and k_{cp} the smallest element in I_{cp} such that $k \leq k_{cp}$.

Let σ^k for all $k \geq 0$ be defined as

$$\sigma^k = \sigma^o / \prod_{m=k_{cp}^+}^{k-1} (1 - \alpha^m(1 - \sigma^m)) \quad \text{for } k \notin I_{cp} \cup I_{cp}^+$$

and

$$\sigma^k = \sigma^o \text{ for } k \in I_{cp} \cup I_{cp}^+.$$

A fairly lengthy but straightforward inductive proof can be used to establish the following results (a detailed proof can be found in González-Lima [9]):

- (i) $\sigma^k \in (0, 1]$ for all $k \geq 0$.
- (ii) $\sigma^k = 1$ if $\alpha^{k-1} = 1$ for $k \notin I_{cp} \cup I_{cp}^+$.
- (iii) $\sigma^{k_{cp}^++1} \leq \sigma^{k_{cp}^++2} \leq \dots \leq \sigma^{k_{cp}} \leq 1$.

From step (3.1) in the algorithm we have

$$(4.1) \quad (x^{k+1})^T z^{k+1} = (x^k)^T z^k (1 - \alpha^k) + n \alpha^k \mu \quad \text{for all } k \geq 0.$$

Then, the remainder of the proof consists in showing that

$$(4.2) \quad \mu = \sigma^k (x^k)^T z^k / n \quad \text{for all } k \geq 0.$$

If $k \in I_{cp} \cup I_{cp}^+$, then $\sigma^k = \sigma^o$, and by the construction of the algorithm we have

$$\mu = \sigma^o \frac{(x^k)^T z^k}{n} = \sigma^k \frac{(x^k)^T z^k}{n}.$$

If $k \notin I_{cp} \cup I_{cp}^+$, assume without loss of generality that $k_{cp}^+ = 0$ and $k = 2$.

From (4.1) we get

$$(4.3) \quad (x^2)^T z^2 = (x^1)^T z^1 (1 - \alpha^1) + n \alpha^1 \mu.$$

Multiplying both sides of (4.3) by σ^2/n , using the definition of σ^2 , and the fact that

$$\mu = \sigma^1 \frac{(x^1)^T z^1}{n},$$

we obtain

$$\mu = \sigma^2 \frac{(x^2)^T z^2}{n}.$$

The use of the same argument in an inductive manner for $k \leq k_{cp}$ completes the proof of (4.2). $\square$

PROPOSITION 4.2. *Let $\mu > 0$ be a parameter selected by Algorithm 3.1 and consider the index set $\mathcal{I}(\mu)$ defined by (3.6). Then the following properties are satisfied for any $k \in \mathcal{I}(\mu)$:*

(i) Δw^k is a descent direction for the merit function

$$f_\mu(x, y, z) = \left\| \frac{XZe - \mu e}{\mu} \right\|_2^2$$

at $w^k = (x^k, y^k, z^k)$.

(ii) $f_\mu(w^{k+1}) \leq (1 - \eta\alpha^k)f_\mu(w^k)$, where $\eta \in (0, 1)$ is chosen as in the algorithm.

(iii) $\left\| \frac{X^k Z^k}{(x^k)^T z^k / n} e - e \right\|_2 \leq f_\mu(w^k)^{1/2}$.

Proof. Since $w^k \in \mathcal{F}^+$ we have

$$f_\mu(w^k) = \frac{1}{\mu^2} \|F_\mu(w^k)\|_2^2 \quad \text{and}$$

$$\nabla f_\mu(w^k) = \frac{2}{\mu^2} (F'_\mu(w^k))^T F_\mu(w^k).$$

Therefore,

$$\nabla f_\mu(w^k)^T \Delta w^k = -\frac{2}{\mu^2} F_\mu^T(w^k) F_\mu(w^k) = -\frac{2}{\mu^2} \|F_\mu(w^k)\|_2^2 < 0$$

and (i) follows.

Part (ii) can be demonstrated in a straightforward manner from condition (3.5) of the algorithm.

Now, let us prove part (iii). Because $\frac{(x^k)^T z^k}{n} e$ is the orthogonal projection of $X^k Z^k e$ over the subspace generated by e we have

$$\left\| X^k Z^k e - \frac{(x^k)^T z^k}{n} e \right\|_2 \leq \|X^k Z^k e - \mu e\|_2 = \mu f_\mu(w^k)^{1/2}$$

for all k .

Hence,

$$\left\| \frac{X^k Z^k e}{(x^k)^T z^k / n} - e \right\|_2 \leq f_\mu(w^k)^{1/2} \left(\frac{\mu}{(x^k)^T z^k / n} \right).$$

By Proposition 4.1, $\mu = \sigma^k (x^k)^T z^k / n$ with $\sigma^k \in (0, 1]$. Therefore,

$$\left\| \frac{X^k Z^k e}{(x^k)^T z^k / n} - e \right\|_2 \leq f_\mu(w^k)^{1/2}.$$

Moreover, if $\sigma^k = 1$ the previous inequality is satisfied as an equality. $\square$

The next lemma is needed for proving that under mild assumptions, condition (3.4) from the algorithm (proximity to the central path) is always satisfied.

LEMMA 4.3. *Let $\mu > 0$ be a parameter selected by Algorithm 3.1 and assume that*

$$\min(X^k Z^k e) / x^k{}^T z^k \geq \gamma / n \text{ for all } k \in \mathcal{I}(\mu) \text{ and some } \gamma \in (0, 1),$$

where $\mathcal{I}(\mu)$ is the index set defined by (3.6). Then,

$$\|(\Delta x^k, \Delta y^k, \Delta z^k)\|_2 \leq M < +\infty$$

for all $k \in \mathcal{I}(\mu)$ and some $M > 0$.

Proof. Suppose $(x^k)^T z^k \geq \epsilon$ for some $\epsilon > 0$ and for all $k \geq 0$. By hypothesis, $x_i^k z_i^k \geq \gamma \frac{(x^k)^T z^k}{n}$, hence,

$$(4.4) \quad x_i^k z_i^k \geq \frac{\gamma \epsilon}{n} \text{ for all } k \geq 0 \text{ and } i = 1, \dots, n.$$

Let $\mathcal{F}_o = \{(x, y, z) \in \mathcal{F} : x^T z \leq (x^0)^T z^0\}$. This set is bounded (see the proof of Lemma 2.1 in Zhang and Tapia [24]). Moreover, by Proposition 4.1, the iterates $(x^k, y^k, z^k) \in \mathcal{F}_o$. Therefore, (4.4) implies that $\{x^k\}$ and $\{z^k\}$ are bounded away from zero for all $k \geq 0$.

Let us denote $w^k = (x^k, y^k, z^k)$ and $\Delta w^k = (\Delta x^k, \Delta y^k, \Delta z^k)$. It follows from the boundedness of the sequences $\{x^k\}$ and $\{z^k\}$ that there exist constants C and M_1 such that

$$\|(F'(w^k))^{-1}\|_2 = \left\| \left(\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ Z^k & 0 & X^k \end{bmatrix} \right)^{-1} \right\|_2 \leq C$$

and

$$(4.5) \quad \|\Delta w^k\|_2 \leq \|(F'(w^k))^{-1}\|_2 \|F_\mu(w^k)\|_2 \leq M_1.$$

Now, suppose that $(x^k)^T z^k \rightarrow 0$. By Theorem 3.2 from Tapia, Zhang, and Ye [18] we have that

$$(4.6) \quad \|(\Delta x^k, \Delta z^k)\|_2 \leq C_1 \sigma^k + C_2 (x^k)^T z^k$$

for some constants $C_1, C_2 > 0$, and σ^k defined as in Proposition 4.1 ($\sigma^k \leq 1$).

Equation (4.6) implies that there exists $N_1 > 0$ such that

$$(4.7) \quad \|(\Delta x^k, \Delta z^k)\|_2 \leq N_1 \text{ for all } k \geq 0.$$

Since we are assuming that $\text{rank}(A) = m$, the matrix AA^T is invertible and

$$(4.8) \quad \Delta y^k = -(AA^T)^{-1} A \Delta z^k.$$

Therefore, (4.7) and (4.8) imply that there exists $N_2 > 0$ such that

$$(4.9) \quad \|\Delta y^k\|_2 \leq N_2 \text{ for all } k \geq 0.$$

The final result follows from (4.5), (4.7), and (4.9). $\square$

The next lemma states that under some natural assumptions the index sets $\mathcal{I}(\mu)$ are finite for any $\mu > 0$ generated by the LSSN algorithm.

LEMMA 4.4. *Suppose in the LSSN algorithm (Algorithm 3.1) the parameter choice $\{\tau^k\}$ has been made so that the following condition is satisfied:*

(c1) $\tau^k \geq \tau$ for all k and some $\tau \in (0, 1)$.

Now, assume that the points generated by the LSSN algorithm satisfy

(a1) $\min(X^k Z^k e)/x^{kT} z^k \geq \gamma/n$ for all k and some $\gamma \in (0, 1)$.

Then, the index sets $\mathcal{I}(\mu)$ defined by (3.6) are finite for any value of μ selected by the algorithm.

Proof. Assume by contradiction that Algorithm 3.1 selects a value of μ such that $\mathcal{I}(\mu)$ is infinite; i.e.,

$$\mathcal{I}(\mu) = \{N, N+1, N+2, \dots\}$$

for some $N > 0$. Under this assumption we will prove that

$$(4.10) \quad f_\mu(w^k) \rightarrow 0,$$

where f_μ is defined by (3.3) and $w^k = (x^k, y^k, z^k)$. It is easily seen that (4.10) implies that condition (3.4) will be satisfied for k large enough, which contradicts the assumption that $\mathcal{I}(\mu)$ is infinite.

Let us call $\bar{\alpha}^k = \min(1, \tau^k \hat{\alpha}^k)$, where

$$\hat{\alpha}^k = \frac{-1}{\min((X^k)^{-1} \Delta x^k, (Z^k)^{-1} \Delta z^k)} \text{ and } \tau^k \in (0, 1).$$

It is known (see the proof of Theorem 3.1 in Zhang, Tapia, and Dennis [25]) that condition (c1) and assumption (a1) imply that $\bar{\alpha}^k$ is bounded away from zero.

If $\alpha^k = \bar{\alpha}^k$ in infinitely many iterations, then by using (ii) of Proposition 4.2 we deduce immediately that (4.10) holds. Now, suppose that $\alpha^k < \bar{\alpha}^k$ for k sufficiently large. It is known (e.g., see Theorem 6.3.3 in Dennis and Schnabel [4] and the consequent discussion) that the backtracking line search used in the LSSN algorithm (Algorithm 3.1) guarantees that

$$(4.11) \quad \frac{\nabla f_\mu(w^k)^T \Delta w^k}{\|\Delta w^k\|_2} \rightarrow 0,$$

where $\Delta w^k = (\Delta x^k, \Delta y^k, \Delta z^k)$. According to the proof of Proposition 4.2 we can write

$$(4.12) \quad \nabla f_\mu(w^k)^T \Delta w^k = \frac{-2}{\mu^2} \|F_\mu(w^k)\|_2^2 = -2f_\mu(w^k),$$

and by virtue of Lemma 4.3, we have that $\|\Delta w^k\|_2 \leq M$ for all k . Therefore, equations (4.11) and (4.12) imply that (4.10) is satisfied in this case as well. This completes the proof of our lemma. $\square$

Using Lemma 4.4 we will show that if the parameters β^j that define the neighborhoods of the central path are chosen so that they converge to zero, then the iteration

sequence generated by Algorithm 3.1 possesses a subsequence converging to the analytic center of the solution set. The next theorem formally states this result.

THEOREM 4.5. *Let $\{(x^k, y^k, z^k)\}$ be generated by the LSSN algorithm (Algorithm 3.1) with parameter choices $\sigma^0 \in (0, 1)$, $\{\beta^k\}$, and $\{\tau^k\}$. Suppose the parameter choices have been made so that the following two conditions are satisfied:*

- (c1) $\tau^k \geq \tau > 0$ for all k and some $\tau \in (0, 1)$.
- (c2) $\beta^k \rightarrow 0$.

Assume

- (a1) $\min(X^k Z^k e)/x^k z^k \geq \gamma/n$ for all k and some $\gamma \in (0, 1)$.

Then, there exists a subsequence of $\{(x^k, y^k, z^k)\}$ that converges to the analytic center of the solution set.

Proof. By assumption (a1), condition (c1), and Lemma 4.4, there exists a subsequence $\{k^j\}$ such that (3.4) is satisfied at iteration k^j ; i.e.,

$$\left\| \frac{X^{k_j} Z^{k_j} e}{\mu^{k_j}} - e \right\|_2 \leq \beta^{k_j}.$$

According to (iii) of Proposition (4.2), we have

$$\left\| \frac{X^{k_j} Z^{k_j} e}{(x^{k_j})^T z^{k_j}/n} - e \right\|_2 \leq (f_\mu(w^{k_j}))^{1/2},$$

and by virtue of condition (c2), it follows that

$$(4.13) \quad \left\| \frac{X^{k_j} Z^{k_j} e}{(x^{k_j})^T z^{k_j}/n} - e \right\|_2 \rightarrow 0.$$

Now, for each $j > 0$, Proposition (4.1) implies that $(x^{k_{j+1}})^T z^{k_{j+1}} \leq (x^{k_j+1})^T z^{k_j+1}$ and $(x^{k_j+1})^T z^{k_j+1} \leq \zeta (x^{k_j})^T z^{k_j}$ for some $\zeta \in (0, 1)$. Therefore,

$$(4.14) \quad (x^{k_j})^T z^{k_j} \rightarrow 0.$$

The remainder of the proof follows from (4.13), (4.14), and Lemma 3.1. $\square$

5. Complexity bound. In this section we address two important issues concerning the LSSN algorithm; namely the centrality assumption in Theorem 4.1 and the complexity of the algorithm. We prove that a modification of the line-search strategy for the LSSN algorithm leads to a modified LSSN algorithm which possesses polynomial complexity and for which the centrality assumption of Theorem 4.1 is always satisfied.

In order to facilitate the exposition below let

$$\mathcal{F}_\gamma = \left\{ (x, y, z) \text{ strictly feasible} : \frac{\min(XZe)}{x^T z/n} \geq \gamma \right\} \text{ for } \gamma \in (0, 1).$$

Clearly if $(x^k, y^k, z^k) \in \mathcal{F}_\gamma$ for all $k \geq 0$, then the sequence $\{(x^k, y^k, z^k)\}$ satisfies assumption (a1) (the centrality assumption) of Theorem 4.1.

For each $\mu > 0$, the modified LSSN algorithm presented in this section considers an exact line search instead of the backtracking strategy performed by the LSSN algorithm. The exact line search is defined so that the iterates generated by the modified LSSN algorithm always belong to the set $\mathcal{F}_\gamma$. Therefore, we demonstrate that the assumption of Theorem 4.1 can always be enforced. Moreover, since the

merit function for the line search is the Euclidean norm proximity measure given by f_μ , a polynomial bound can be derived for the number of iterations required by the algorithm to satisfy the proximity condition to the central path given by (3.4). Hence, polynomial complexity is attainable.

The following is a description of the modified LSSN algorithm.

ALGORITHM 5.1 (modified long-step shrinking-neighborhood (LSSN) algorithm). *Consider a strictly feasible point $w^0 = (x^0, y^0, z^0)$, $\sigma^0 \in (0, 1)$, and $\beta^0 \in (0, 1)$. Choose $\gamma \in (0, 2 - \sqrt{2}]$ such that $\gamma \leq \frac{\min(X^0 Z^0 e)}{(x^0)^T z^0 / n}$.*

Do until convergence

- (0) Set $k = 0$.
- (1) Set $\mu^k = \sigma^0 (x^k)^T z^k / n$.
- (2) Set $\mu = \mu^k$ and $\beta = \beta^k$.
- (3) Denote $w^{k,0} = w^k$.
- (4) Set $l = 0$.
- (5) Set $w = (x, y, z) = w^{k,l}$.
- (6) Check proximity to the central path: If

$$(5.1) \quad \|XZe/\mu - e\|_2 \leq \beta,$$

go to (10).

- (7) Compute new iterate:

(7.a) Solve the following system for $\Delta w = (\Delta x, \Delta y, \Delta z)$:

$$F'(x, y, z) \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = -F(x, y, z) + \mu \begin{pmatrix} 0 \\ 0 \\ e \end{pmatrix}.$$

(7.b) Line search: Find the solution $\bar{\theta}$ of

$$(5.2) \quad \min_{w + \theta \Delta w \in \mathcal{F}_\gamma} f(\theta),$$

where $f(\theta) = \|X(\theta)z(\theta)/\mu - e\|_2$.

(7.c) Form the iterate $w^+ = w + \bar{\theta} \Delta w$.

- (8) Set $w^{k,l+1} = w^+$ and $\theta^{k,l} = \bar{\theta}$.
- (9) Set $l := l + 1$, and go to (5).
- (10) Set $l^k = l$ and $w^{k+1} = w^{k,l^k}$.
- (11) Decrease β -neighborhood: Choose $\beta^{k+1} \leq \beta^k$.
- (12) Set $k := k + 1$, and go to (1).

Observe that in this modification the iterations where the β -neighborhoods of the central path are reached are counted separately from the other iterations.

In order to obtain a polynomial complexity bound for the modified LSSN algorithm we need to derive an upper bound for the number of iterations required by the algorithm to satisfy the central-path proximity condition given in step (6). The idea for bounding the number of iterations required by the algorithm to approach the central path is to study the line-search problem (5.2) and to find an upper bound for its objective function value. The next lemmas address these issues.

LEMMA 5.1. *Let $\gamma \in (0, 1)$, $w = (x, y, z) \in \mathcal{F}_\gamma$, and $\mu \leq x^T z / n$.*

Also let $\Delta w = (\Delta x, \Delta y, \Delta z)$ denote the solution of the system

$$F'(x, y, z) \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = -F(x, y, z) + \mu \begin{pmatrix} 0 \\ 0 \\ e \end{pmatrix}.$$

Consider $\phi = \frac{1}{\mu} \|\Delta X \Delta Z e\|_2$, where $\Delta X = \text{diag}(\Delta x)$, $\Delta Z = \text{diag}(\Delta z)$.

Then, $w + \theta \Delta w \in \mathcal{F}_\gamma$ for $0 \leq \theta \leq \theta^* = \min(1, (1 - \gamma)/\phi)$.

Proof. Let $w(\theta) = (x(\theta), y(\theta), z(\theta)) = w + \theta \Delta w$. Then, we have that

$$(5.3) \quad X(\theta)z(\theta) = (1 - \theta)Xz + \theta\mu e + \theta^2 \Delta X \Delta z.$$

Since w is strictly feasible, $\Delta x^T \Delta z = 0$ and from (5.3) we obtain

$$(5.4) \quad x(\theta)^T z(\theta)/n = (1 - \theta)(x^T z/n) + \theta\mu.$$

For any $0 \leq \theta \leq 1$ we deduce that

$$X(\theta)z(\theta) \geq \gamma(1 - \theta)(x^T z/n)e + \theta\mu e - \theta^2 \phi \mu e,$$

which implies

$$(5.5) \quad X(\theta)z(\theta) \geq \gamma(x(\theta)^T z(\theta)/n)e + \theta[(1 - \gamma) - \theta\phi]\mu e.$$

It follows from (5.5) that if $0 \leq \theta \leq \theta^* = \min(1, (1 - \gamma)/\phi)$ then $\frac{\min(X(\theta)Z(\theta)e)}{x(\theta)^T z(\theta)/n} \geq \gamma$.

Now, let us show that

$$(5.6) \quad x(\theta) > 0, z(\theta) > 0 \quad \text{for all } 0 \leq \theta \leq \theta^* = \min(1, (1 - \gamma)/\phi).$$

For $\theta = 0$ we have $w(0) = w$, and (5.6) is satisfied because of the hypothesis that $w \in \mathcal{F}_\gamma$.

Let us suppose by contradiction that (5.6) is not satisfied for all $\theta \in [0, \theta^*]$. Then there exists $\hat{\theta} \in [0, \theta^*]$ and $\hat{i} \in \{1, 2, \dots, n\}$ such that either $x_{\hat{i}}(\hat{\theta}) = 0$ or $z_{\hat{i}}(\hat{\theta}) = 0$. But then from (5.5) we get

$$(5.7) \quad 0 \geq \gamma(x(\hat{\theta})^T z(\hat{\theta})/n) + \hat{\theta}[(1 - \gamma) - \hat{\theta}\phi]\mu.$$

From (5.4) we deduce that $x(\hat{\theta})^T z(\hat{\theta})/n \geq \hat{\theta}\mu$ so that (5.7) gives the contradiction $0 \geq \gamma\hat{\theta}\mu$. $\square$

Lemma 5.1 states that the minimization problem (5.2) gives a lower value of the objective function than does the problem

$$(5.8) \quad \min_{0 \leq \theta \leq \theta^*} f(\theta).$$

The next lemma uses the result obtained from Lemma 5.1 and the optimization problem (5.8) in order to obtain an upper bound for the optimal value of (5.2). This upper bound will be used later for bounding the number of iterations needed to satisfy (5.1), i.e., the number of iterations needed for entering the β -neighborhood of the central path.

LEMMA 5.2. *Let $\gamma \in (0, 2 - \sqrt{2}]$, $w = (x, y, z) \in \mathcal{F}_\gamma$, $\mu \leq x^T z/n$, and $\rho = \|(Xz - \mu e)/\mu\|_2$.*

The optimal value of problem (5.2) is bounded above by

$$\rho^* = \rho \max \left\{ 1 - \frac{\gamma}{2\sqrt{2}(1 - \gamma)}, 1 - \frac{\sqrt{2}\gamma(1 - \gamma)}{\rho^2} \right\}.$$

Proof. From (5.3) we deduce that

$$\begin{aligned} f(\theta) &= \left\| (1-\theta) \frac{Xz}{\mu} + \theta e + \theta^2 \frac{\Delta X \Delta z}{\mu} - e \right\|_2 \\ &= \left\| (1-\theta) \left(\frac{Xz}{\mu} - e \right) + \theta^2 \frac{\Delta X \Delta z}{\mu} \right\|_2. \end{aligned}$$

As in the previous lemma, let us denote $\phi = \frac{1}{\mu} \|\Delta X \Delta z\|_2$. Then for any $0 \leq \theta \leq 1$ we obtain the inequality

$$(5.9) \quad f(\theta) \leq (1-\theta) \left\| \frac{Xz}{\mu} - e \right\|_2 + \theta^2 \phi.$$

Let $\rho = \|(Xz - \mu e)/\mu\|_2$ and $\rho(\theta) = (1-\theta)\rho + \theta^2 \phi$. From (5.9) we have

$$(5.10) \quad f(\theta) \leq \rho(\theta).$$

Let

$$(5.11) \quad \rho^+ = \min_{0 \leq \theta \leq \theta^*} \rho(\theta).$$

It is clear from (5.10) and Lemma 5.1 that the optimal value of problem (5.2) is bounded above by ρ^+ . The remainder of the proof consists in finding an upper bound to ρ^+ .

Note that $\rho(0) = f(0) = \rho$. The unconstrained minimum of $\rho(\theta) = (1-\theta)\rho + \theta^2 \phi$ is obtained for $\hat{\theta} = \rho/(2\phi)$. We consider four cases.

(i) If $\rho \leq 2(1-\gamma)$ and $(1-\gamma)/\phi \leq 1$, then $\rho/(2\phi) = \hat{\theta} \leq \theta^* = (1-\gamma)/\phi$ and consequently

$$(5.12) \quad \rho^+ = \rho(\hat{\theta}) = \left(1 - \frac{\rho}{4\phi}\right) \rho.$$

We want to find an upper bound for ρ^+ in terms of ρ and γ . To this effect we use the inequality

$$\|\Delta X \Delta z\|_2 \leq \frac{1}{2\sqrt{2}} \|(XZ)^{-1/2} (Xz - \mu e)\|^2,$$

whose proof can be found in [16]. According to the hypothesis of our lemma, $w = (x, y, z) \in \mathcal{F}_\gamma$ and $\mu \leq x^T z/n$ which leads us to the inequality

$$(5.13) \quad \phi \leq \frac{\rho^2}{2\sqrt{2}\gamma}.$$

Substituting into (5.12) and using the fact that $\rho \leq 2(1-\gamma)$ we obtain the following bound for case (i):

$$(5.14) \quad \rho^+ \leq \left(1 - \frac{\gamma}{2\sqrt{2}(1-\gamma)}\right) \rho.$$

(ii) On the other hand if $\rho \leq 2(1-\gamma)$ and $(1-\gamma)/\phi > 1$, then $\theta^* = 1$ and we may have either $\hat{\theta} \leq \theta^* = 1$, in which case (5.12) and (5.14) hold, or $\hat{\theta} > \theta^* = 1$ and then

$$(5.15) \quad \rho^+ = \rho(1) = \phi < \rho/2.$$

Note that since $\gamma \leq 2 - \sqrt{2}$, then $\rho/2$ is less than the quantity on the right-hand side of (5.14) so that (5.14) holds whenever $\rho \leq 2(1-\gamma)$.

(iii) Now if we suppose that $\rho > 2(1-\gamma)$ and $(1-\gamma)/\phi \leq 1$, then $\rho/(2\phi) = \hat{\theta} > \theta^* = (1-\gamma)/\phi$, which implies

$$(5.16) \quad \rho^+ = \rho(\theta^*) = \left(1 - \frac{1-\gamma}{\phi} + \frac{(1-\gamma)^2}{\phi\rho}\right) \rho < \left(\frac{1-\gamma}{2\phi}\right) \rho.$$

Using (5.13) we obtain the upper bound

$$(5.17) \quad \rho^+ \leq \left(1 - \frac{\sqrt{2}\gamma(1-\gamma)}{\rho^2}\right) \rho.$$

(iv) Finally, if we assume that $\rho > 2(1-\gamma)$ and $(1-\gamma)/\phi > 1$, then $\rho/(2\phi) = \hat{\theta} > \theta^* = 1$, and therefore (5.15) holds in this case as well. But then so does (5.14) and (5.17).

In conclusion, we have shown that (5.14) holds whenever $\rho \leq 2(1-\gamma)$, while (5.17) holds whenever $\rho > 2(1-\gamma)$. We end the proof of our lemma by noting that the right-hand side of (5.17) is greater than the right-hand side of (5.14) if and only if $\rho > 2(1-\gamma)$. $\square$

LEMMA 5.3. *Let $\{(x^{k,l}, y^{k,l}, z^{k,l})\}$ and $\{\mu^k\}$ be generated by the modified LSSN algorithm with the parameter choice $\{\beta^k\}$.*

For each $k \geq 0$, let l^k denote the iterate such that

$$\|X^{k,l^k} z^{k,l^k} / \mu^k - e\|_2 \leq \beta^k.$$

Then l^k is well defined with

$$l^0 = O(n^2/\gamma) \text{ and } l^k = O(n/\gamma) + O(\log \beta^k) \text{ for } k \geq 1.$$

Proof. Let $\rho^{k,l} = \|X^{k,l} z^{k,l} / \mu^k - e\|_2$. From Lemma 5.2 we have that

$$(5.18) \quad \rho^{k,l+1} \leq \left(1 - \frac{\gamma}{2\sqrt{2}(1-\gamma)}\right) \rho^{k,l} \text{ if } \rho^{k,l} \leq 2(1-\gamma)$$

and

$$(5.19) \quad \rho^{k,l+1} \leq \left(1 - \frac{\sqrt{2}\gamma(1-\gamma)}{(\rho^{k,l})^2}\right) \rho^{k,l} \text{ if } \rho^{k,l} > 2(1-\gamma).$$

Using (5.19) and the fact that $\rho^{k,j+1} \leq \rho^{k,j}$, we obtain the relation

$$\rho^{k,l+1} \leq \left(1 - \frac{\sqrt{2}\gamma(1-\gamma)}{\rho^{k,l} \rho^{k,0}}\right) \rho^{k,l} = \left(\rho^{k,l} - \frac{\sqrt{2}\gamma(1-\gamma)}{\rho^{k,0}}\right).$$

It follows that

$$(5.20) \quad \rho^{k,l} \leq \rho^{k,0} - l \frac{\sqrt{2}\gamma(1-\gamma)}{\rho^{k,0}}.$$

Therefore, $\rho^{k,l} \leq 2(1-\gamma)$ in at most

$$(5.21) \quad \bar{l}^k = \frac{(\rho^{k,0} - 2(1-\gamma))\rho^{k,0}}{\sqrt{2}\gamma(1-\gamma)}$$

steps.

From (5.18)

$$\rho^{k,\bar{l}^k+l} \leq \left(1 - \frac{\gamma}{2\sqrt{2}(1-\gamma)}\right)^l \rho^{k,\bar{l}^k} \leq (1-\alpha^k)^l 2(1-\gamma),$$

where $\alpha = \gamma/(2\sqrt{2}(1-\gamma))$.

It follows that $\rho^{k,\bar{l}^k+l} \leq \beta^k$ in at most

$$(5.22) \quad \hat{l}^k = \frac{\log(\frac{\beta^k}{2(1-\gamma)})}{\log(1-\alpha)}$$

steps.

The following observations are in order. If $\rho^{k,0} < 2(1-\gamma) \leq \beta^k$, then $l^k = \bar{l}^k = \hat{l}^k = 0$. If $2(1-\gamma) > \beta^k$, then $\bar{l}^k = 0$ and $\hat{l}^k$ is given by (5.22).

On the other hand, if $\rho^{k,0} > 2(1-\gamma)$ and $2(1-\gamma) \leq \beta^k$, we have that $l^k = \bar{l}^k$ as defined by (5.21) and $\hat{l}^k = 0$.

Hence, l^k is well defined and from (5.21) and (5.22) we obtain that $l^k \leq \bar{l}^k + \hat{l}^k$.

The rest of the proof is devoted to finding bounds for $\rho^{k,0}$.

For any $0 \leq \mu \leq (x^0)^T z^0/n$, it is clear that

$$(5.23) \quad \|X^0 z^0 - \mu e\|_2^2 \leq \|X^0 z^0\|_2^2 \leq ((x^0)^T z^0)^2.$$

Let $\mu^0 = \sigma^0 (x^0)^T z^0/n$. It follows from (5.23) that $\rho^{0,0} \leq n/\sigma^0$.

According to (5.21), this implies that

$$(5.24) \quad \bar{l}^0 = O(n^2/\gamma).$$

If β^0 is independent of n , say $\beta^0 = \frac{1}{4}$, then

$$(5.25) \quad \hat{l}^0 = O(1)$$

so that

$$(5.26) \quad l^0 = O(n^2/\gamma).$$

Now,

$$\begin{aligned} (\rho^{k,0})^2 &= \|X^k z^k / \mu^k - e\|_2^2 = \frac{1}{(\mu^k)^2} \|X^k z^k - \mu^k e\|_2^2 \\ &= \frac{1}{(\mu^k)^2} (\|X^k z^k - (x^{kT} z^k/n) e\|_2^2 + (\mu^k - x^{kT} z^k/n)^2 \|e\|_2^2). \end{aligned}$$

By definition of the modified LSSN algorithm,

$$(5.27) \quad \|X^k z^k - \mu^{k-1} e\|_2 \leq \beta^{k-1} \mu^{k-1} \text{ for each } k \geq 1,$$

and because $(x^{kT} z^k/n)e$ is the orthogonal projection of $X^k z^k$ over the subspace generated by e , we have

$$(5.28) \quad \|X^k z^k - (x^{kT} z^k/n)e\|_2 \leq \|X^k z^k - \mu^{k-1}e\|_2.$$

Since $\mu^k = \sigma^0 x^{kT} z^k/n$, we obtain

$$\begin{aligned} (\rho^{k,0})^2 &\leq \frac{1}{(\mu^k)^2} ((\beta^{k-1})^2 (x^{kT} z^k/n)^2 + n(\mu^k - x^{kT} z^k/n)^2) \\ &= \frac{1}{(\mu^k)^2} ((\beta^{k-1})^2 (x^{kT} z^k/n)^2 + n(1 - \sigma^0)^2 (x^{kT} z^k/n)^2). \end{aligned}$$

It follows that

$$(5.29) \quad (\rho^{k,0})^2 \leq \frac{(\beta^*)^2 + (1 - \sigma^0)^2 n}{(\sigma^0)^2},$$

where $\beta^* = \max_{k \geq 0} \beta^k$.

Finally, from (5.21), (5.22), (5.29), and $l^k \leq \bar{l}^k + \hat{l}^k$ we deduce that $l^k = O(n/\gamma) + O(\log \beta^k)$ for $k \geq 1$. $\square$

The previous lemma gives an upper bound for the number of iterations required by the modified LSSN algorithm to enter the β -neighborhood of the central path. The polynomial complexity bound for the algorithm is obtained by assuming a certain rate of decrease for β^k and by finding a bound on the number of iterations required by the algorithm to reduce the gap to a given tolerance. This is done in the next theorem.

THEOREM 5.4. *Let $\{(x^k, y^k, z^k)\}$ and $\{\mu^k\}$ be generated by the modified LSSN algorithm with $\{(x^0, y^0, z^0)\}$ a strictly feasible point and the parameter choices $\{\beta^k\}$.*

Choose $\epsilon > 0$ and $\sigma^0 \in (0, 1)$ such that $\sigma^0(\beta^* + 1) < 1$, where $\beta^* = \max_{k \geq 0} \beta^k$. Then,

- (i) $x^{kT} z^k \leq \epsilon$ in at most k_ϵ steps with $k_\epsilon = O(\log(x^0)^T z^0 / \epsilon)$.
- (ii) Let β^k be chosen as $\beta^k = (\beta^0)^k$. Then, the modified LSSN algorithm iteration complexity is

$$O(n^2/\gamma) + O\left(\left\lceil n/\gamma + \log \frac{(x^0)^T z^0}{\epsilon} \right\rceil \log \frac{(x^0)^T z^0}{\epsilon}\right).$$

- (iii) Let β^k be chosen as $\beta^k = \frac{1}{2^{(k+1)}}$. Then, the modified LSSN algorithm iteration complexity is

$$O(n^2/\gamma) + O\left(\left\lceil n/\gamma + \log \left(\log \frac{(x^0)^T z^0}{\epsilon}\right) \right\rceil \log \left(\frac{(x^0)^T z^0}{\epsilon}\right)\right).$$

Proof. From (5.27) we have that $x_i^{k+1} z_i^{k+1} \leq (1 + \beta^k) \mu^k = (1 + \beta^k) \sigma^0 x^{kT} z^k/n$ for all $k \geq 0$. Therefore, $x^{kT} z^k \leq \eta^k (x^0)^T z^0$ where $\eta = (1 + \beta^*) \sigma^0$ with $\beta^* = \max_{k \geq 0} \beta^k$ and $\eta < 1$ by hypothesis. Hence, $x^{kT} z^k \leq \epsilon$ in at most $k_\epsilon = O(\log(x^0)^T z^0 / \epsilon)$.

From Lemma 5.3

$$(5.30) \quad \sum_{k=1}^{k_\epsilon} l^k = O\left(\frac{n}{\gamma} \log \left(\frac{(x^0)^T z^0}{\epsilon}\right)\right) + O\left(\log \left(\frac{(x^0)^T z^0}{\epsilon}\right)^2\right) \text{ if } \beta^k = (\beta^0)^k$$

and

$$(5.31) \quad \sum_{k=1}^{k_\epsilon} l^k = O\left(\frac{n}{\gamma} \log\left(\frac{(x^0)^T z^0}{\epsilon}\right)\right) + O\left(\log\left(\frac{(x^0)^T z^0}{\epsilon}\right) \log\left(\log\left(\frac{(x^0)^T z^0}{\epsilon}\right)\right)\right)$$

$$\text{if } \beta^k = \frac{1}{2(k+1)}.$$

Items (ii) and (iii) follow from (5.30), (5.31), and the fact that $l^0 = O(n^2/\gamma)$ (see Lemma 5.3). $\square$

COROLLARY 5.5. *Consider problem (2.1) and assume the entries of the data A, b, c are integers with L being the bit length of that input data.*

Let $\{(x^0, y^0, z^0)\}$ be a strictly feasible point such that $(x^0)^T z^0 \leq 2^{2L}$.

Then, iteration complexity of the modified LSSN algorithm is

$$O(n^2/\gamma) + O([n/\gamma + L]L) \quad \text{if } \beta^k = C\beta^{k-1} \text{ for } k \geq 1 \text{ and for some } C \in (0, 1)$$

and

$$O(n^2/\gamma) + O([n/\gamma + \log(L)]L) \quad \text{if } \beta^k = O\left(\frac{1}{k}\right) \text{ for } k \geq 1.$$

Proof. Let $\epsilon = 2^{-2L}$. The proof follows directly from Theorem 5.4 since $\log \frac{(x^0)^T z^0}{\epsilon} \leq 4L$. $\square$

6. Numerical experience. In this section we discuss the numerical results obtained from applying the LSSN algorithm to a subset of the NETLIB test problems. We compare the performance of the LSSN algorithm with the performance of the Kojima–Mizuno–Yoshise algorithm and the Mizuno–Todd–Ye algorithm.

Our preliminary numerical experimentation showed that the LSSN algorithm as presented in section 3 (Algorithm 3.1) seems to perform better than the modified LSSN algorithm in terms of number of iterations. Therefore, the results in the present section use the LSSN algorithm without modification (Algorithm 3.1). At this juncture we believe that the line-search modification introduced in section 5 has more of a theoretical value than a practical value. However, more research will be done on this issue.

6.1. Implementation issues. The experiments were performed in 64 bit arithmetic on a Sun 4/model 670-120 using codes implemented in MATLAB.

6.1.1. Stopping criteria. The algorithm is terminated when

$$\max\left(\frac{|c^T x^k - b^T y^k|}{1 + |b^T y^k|}, \frac{\|Ax^k - b\|_1}{1 + \|x\|_1}, \frac{\|A^T y^k + z^k - c\|_1}{1 + \|y^k\|_1 + \|z^k\|_1}, \frac{\|X^k z^k - e\|_2}{(x^k)^T z^k / n}\right) \leq 10^{-8}$$

or when the number of iterations reaches 200.

6.1.2. Parameter choices. The line-search strategy (backtracking) defined in step (4) of the LSSN algorithm was implemented using the value $\eta = 10^{-4}$ and a fixed value $\rho = 1/2$.

In all problems, the parameters τ^k were chosen as $\tau^k = 1 - \min\{0.05, 0.05(x^k)^T z^k\}$. Therefore, $\tau^k = 0.95$ far away from the solution set and is closer to 1 when the duality gap is small.

The step lengths for the Mizuno–Todd–Ye algorithm were obtained following Ye et al. [20].

The parameters β^j were chosen, in general, as $\beta^{k+1} = (\beta^k)^2$. Some minor variations, based on heuristics, were considered in choosing β^j when the iterates were close to the solution set. In the numerical results presented in this section, $\beta = \beta^0 = 0.25$ for the Mizuno–Todd–Ye and LSSN algorithms. The choice of $\beta^0 = 0.25$ was made so that we could compare the Mizuno–Todd–Ye algorithm with the LSSN algorithm. Our experience has shown that numerical efficiency for the LSSN algorithm can be substantially improved by choosing β significantly larger than 0.25.

6.1.3. Starting iterates. The starting points for the LSSN and Kojima–Mizuno–Yoshise algorithms are obtained following Lustig, Marsten, and Shanno [12] and are not necessarily feasible.

The algorithms generate a sequence of iterates that approach feasibility and have the property that the duality gap goes to zero. For the LSSN algorithm, the infeasibility of the starting iterates should be considered a computational convenience since in all the problems tested when the initial β -neighborhood of the central path is entered for the very first time the iterates are already feasible. Thus, equivalent experimentation could have been done starting the algorithm from the strictly feasible iterates obtained in the process.

A very interesting feature of the LSSN algorithm is that it provides a way of computing a point in a β -neighborhood of the central path for any $\beta \in (0, 1)$. In this work we used this feature of the LSSN algorithm for computing the starting point for the Mizuno–Todd–Ye algorithm.

6.2. Numerical results. Our numerical results are summarized in the following three tables. Table 6.1 studies the effect of the choice of the parameter σ^0 in the performance of the LSSN algorithm. Table 6.2 compares the performance of the LSSN and the Mizuno–Todd–Ye algorithms. The starting point is the same for both algorithms and the iterations are counted from that point. Table 6.3 shows the performance of the Kojima–Mizuno–Yoshise algorithm for fixed values of the centering parameters σ^k .

Observe that the worst performance of the LSSN algorithm is obtained for very small or very large values of σ^0 . On one hand, if the parameter σ^0 is small (e.g., $\sigma^0 \leq 0.0001$ for most of the problems we considered), the number of iterations exceeds the allowable maximum number of 200 iterations. This large number of iterations is due to the fact that the iterates approach the solution set very fast but are still far away from the central path. Because of the proximity of the iterates to the solution set, the Jacobian matrix is very ill conditioned and the step lengths become very small. On the other hand, comparing the values of σ^0 for which the algorithm converges, it can be observed that the closer the value of the parameter σ^0 is to 1, the stronger is the adherence of the iterates to the central path and the slower is the performance of the algorithm (as should be expected from the theory). Observe from Table 6.1 that σ^0 equal to 0.01 seems to be a good choice to use in implementations.

Notice that the number of iterations required to enter the initial β -neighborhood does not seem to follow, in general, a constant pattern with respect to the values of σ^0 . However, when the line-search strategy is activated, the larger the value of σ^0 , the larger the number of times the step must be decreased (by backtracking). We believe this behavior reflects the fact that the larger the value of σ^0 , the more difficult it is for the LSSN algorithm to obtain appropriate decrease in the merit function since a higher priority is placed on centrality.

For most of the problems tested, the Mizuno–Todd–Ye algorithm requires a larger number of linear systems to be solved to converge than does the LSSN algorithm, as

TABLE 6.1
Study of the effect of different choices of the parameter σ^0 for the LSSN algorithm.

PROBLEM	σ^0	k ¹	kcp ²	ls ³	fc ⁴
AFIRO	0.00001	20	15	0	2
	0.0001	21	13	0	3
	0.001	22	11	0	3
	0.01	20	7	0	3
	0.1	25	8	0	5
	0.5	65	8	0	18
BLEND	0.00001	NC ⁵	–	–	–
	0.0001	NC	–	–	–
	0.001	NC	–	–	–
	0.01	30	12	0	3
	0.1	40	12	2	6
	0.3	71	25	34	11
SCSD1	0.00001	NC	–	–	–
	0.0001	NC	–	–	–
	0.001	NC	–	–	–
	0.01	25	4	0	4
	0.1	34	3	0	7
	0.5	76	3	0	21
SHARE2B	0.00001	28	21	0	2
	0.0001	NC	–	–	–
	0.001	28	13	0	3
	0.01	33	12	0	3
	0.1	41	13	0	6
	0.5	121	57	145	17
SCTAP1	0.00001	NC	–	–	–
	0.0001	NC	–	–	–
	0.001	NC	–	–	–
	0.01	44	10	0	4
	0.1	61	18	17	7
	0.5	123	39	81	20
LOTFI	0.00001	NC	–	–	–
	0.0001	NC	–	–	–
	0.001	NC	–	–	–
	0.01	96	61	86	4
	0.1	109	63	87	8
	0.5	180	70	155	21
SCAGR7	0.00001	33	21	0	2
	0.0001	35	18	0	3
	0.001	32	13	0	3
	0.01	36	11	0	4
	0.1	46	10	0	7
	0.5	77	10	0	16
SCAGR25	0.00001	43	32	0	2
	0.0001	40	26	0	3
	0.001	40	20	0	3
	0.01	37	14	0	3
	0.1	45	11	0	6
	0.5	77	11	0	16

¹Number of iterations.²Number of iterations taken to enter the initial β -neighborhood of the central path.³Number of times the line-search strategy decreases the step length.⁴Number of times μ is decreased.⁵The allowable maximum number of 200 iterations is reached.

TABLE 6.2

Comparison between the Mizuno–Todd–Ye and LSSN algorithms in terms of number of linear systems solved.

PROBLEM	M-T-Y	LSSN	σ^0
AFIRO	16	11	0.001
	28	13	0.01
	32	17	0.1
BLEND	52	18	0.01
	54	28	0.1
	54	69	0.5
SCSD1	62	21	0.01
	70	73	0.5
SHARE2B	18	15	0.00001
	42	15	0.001
	58	21	0.01
	62	28	0.1
	64	64	0.5
SCTAP1	122	43	0.1
	90	34	0.01
LOTFI	106	46	0.1
SCAGR7	52	19	0.001
	84	67	0.5
SCAGR25	80	34	0.1
SDSD6	NC ¹	47	0.1

¹The allowable maximum number of 200 iterations is reached.

TABLE 6.3

Performance of the Kojima–Mizuno–Yoshise algorithm.

PROBLEM	Number of iterations	σ^k
AFIRO	NC ¹	0.3
	39	0.5
	105	0.8
BLEND	NC	0.3
	NC	0.5
	NC	0.8
SHARE2B	NC	0.3
	NC	0.5
	NC	0.8
LOTFI	NC	0.5
	NC	0.8
SCAGR7	39	0.3
	56	0.5
	146	0.8
SCAGR25	41	0.3
	53	0.5
	138	0.8

¹The allowable maximum number of 200 iterations is reached.

can be observed from Table 6.2. Recall that the cost per iteration of the Mizuno–Todd–Ye algorithm is twice that of the LSSN algorithm. Actually, for many of the problems the Mizuno–Todd–Ye algorithm also requires a larger number of iterations to converge than does the LSSN algorithm. This result seems to be a consequence of the long steps performed by the LSSN algorithm. An illustration of the behavior of the Mizuno–Todd–Ye and LSSN algorithms with respect to closeness to the central

path and the duality gap is given by Figures 6.1 and 6.2 for problem SHARE2B.

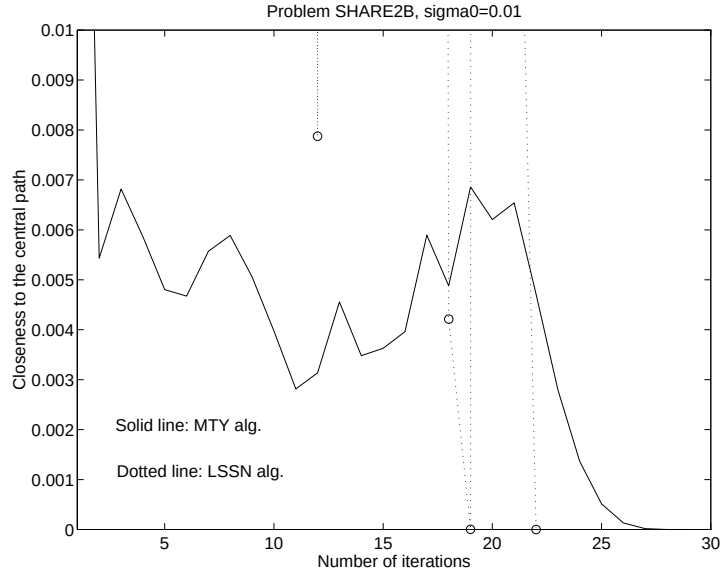


FIG. 6.1. Comparison of the Mizuno-Todd-Ye and LSSN algorithms' behaviors for small value of σ^0 .

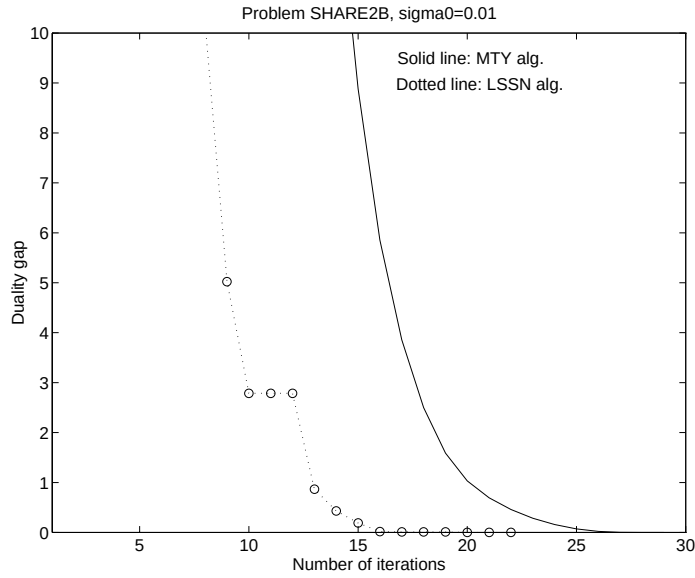


FIG. 6.2. Comparison of the Mizuno-Todd-Ye and LSSN algorithms' behaviors for small value of σ^0 .

Finally, Table 6.3 shows that the Kojima-Mizuno-Yoshise algorithm seems not to be converging for several of the problems considered. This is a significant disadvantage when compared with the LSSN and Mizuno-Todd-Ye algorithms. The lack of convergence of the Kojima-Mizuno-Yoshise algorithm is due to the fact that the

iterates do not stay in or return to a β -neighborhood of the central path after such a neighborhood is reached. The poor performance of the Kojima–Mizuno–Yoshise algorithm is a consequence of the fact that our stopping criteria reflects convergence to the analytic center, and apparently this algorithm does not in general converge to the analytic center.

In the problems where the Kojima–Mizuno–Yoshise algorithm converges, the number of iterations increases as the value of the parameter σ^k increases, as should be expected. The smallest number of iterations required by the Kojima–Mizuno–Yoshise algorithm to converge is always larger than the smallest number of iterations required by the LSSN algorithm to converge for the same problem. It is very interesting to note that for those problems where the Kojima–Mizuno–Yoshise algorithm converges, the line-search strategy is never activated when applying the LSSN algorithm, i.e., they seemed to be rather easy problems. In fact, we have found out that those problems share the feature of having a unique primal solution.

7. Conclusions. In this work, a modification of the Kojima–Mizuno–Yoshise primal-dual algorithm is proposed. The goals of approaching the central path and the solution set are combined to design an algorithm which effectively computes the analytic-center solution. The iterates generated by the LSSN algorithm proposed in this research approach the solution set while approaching the central path. The approach to the central path is done in such a way that we attempt to take full Newton steps (long steps) as much as possible. Two aspects of the algorithm facilitate this objective. First, when the duality gap is large we choose β -neighborhoods that are large. Second, once we reach a β -neighborhood our subsequent step is not required to stay in a β -neighborhood, i.e., is not required to stay near the central path. However, as might be expected from the Gonzaga and Tapia [8] convergence theory for the Mizuno–Todd–Ye algorithm, near the solution when the duality gap is small, full Newton steps actually stay close to the central path and often fall in our chosen β -neighborhoods even though the β -neighborhoods are shrinking.

Let us now briefly make some comments concerning shrinking β -neighborhoods versus fixed β -neighborhoods as employed in the Mizuno–Todd–Ye algorithm. It is clear that far from the solution when the duality gap is large it is not computationally effective to enforce excessive fidelity to the central path; hence, we should allow for large β -neighborhoods in this situation. Our convergence theory given in section 4 for the algorithm described in section 3 is based on the Zhang–Tapia [24] sufficient condition for convergence to the analytic center. This theory requires the satisfaction of condition (3.2), which says that for the shrinking β -neighborhoods we must have the sequence of β s converging to zero. It is also worth noting that our polynomial bound complexity theory given in section 5 requires the sequence of β s to converge to zero. Quite recently Bonnans and Gonzaga [1] derived an elegant theory giving conditions that are sufficient for convergence to the analytic center. Their theory is less restrictive than the Zhang–Tapia [24] theory and does not require the sequence of β s to converge to zero. Their theory suggests that in our algorithm, near the solution when the duality gap is small, β can be held fixed. Very recently, while this paper was in review, we performed some preliminary numerical investigations with a version of our algorithm which left β fixed near the solution. The results were quite satisfactory. Hence, it seems as if the Bonnans–Gonzaga theory could be used to establish convergence to the analytic center for a version of our algorithm that allows fixed β when the duality gap is small. Moreover, the Bonnans–Potra [2] convergence theory, which extends the Bonnans–Gonzaga theory to infeasible iterates, could be

used to analyze our algorithm in the case of infeasible iterates. Recall, as we previously mentioned, that in all our numerical experiments feasibility was established in the first β -neighborhood pass and the iterates remained feasible in subsequent β -neighborhood passes. Hence, the theory presented in section 4 is applicable.

A very interesting feature of the LSSN algorithm is that it gives a practical way of finding a point in a neighborhood of the central path. Hence, the LSSN algorithm can be used for finding a starting point for the Mizuno–Todd–Ye algorithm, and comparisons of the performances of the generic Kojima–Mizuno–Yoshise, Mizuno–Todd–Ye, and LSSN algorithms have been established.

Numerical results showed that the generic Kojima–Mizuno–Yoshise algorithm is not adequate for computing the analytic-center solution even if the centering parameters σ^k are chosen close to one and the iterates reach the central path at some iteration. The Mizuno–Todd–Ye algorithm and the LSSN algorithm are, in general, both capable of computing the analytic center. The LSSN algorithm compares favorably to the Mizuno–Todd–Ye algorithm regarding total cost. In this work we proposed the use of the LSSN algorithm for finding the starting point for the Mizuno–Todd–Ye algorithm, but different approaches for obtaining this initial point may lead to better performance of the Mizuno–Todd–Ye algorithm. The performance of the LSSN algorithm strongly depends on the choice of the parameter σ^0 . The performance of the Mizuno–Todd–Ye algorithm depends on the starting point and does not involve parametric choices, thus, in this sense, the Mizuno–Todd–Ye algorithm is less complicated than the LSSN algorithm. However, it was our experience that while in theory the Mizuno–Todd–Ye algorithm is guaranteed to converge to the analytic center, in practice this is not necessarily the case (obviously due to the ill conditioning encountered near the solution set).

Acknowledgments. The authors would like to thank Rongqin Sheng for carefully reading the manuscript and suggesting several improvements. They also thank Amr El-Bakry for numerous discussions and two anonymous referees for constructive comments.

REFERENCES

- [1] J. F. BONNANS AND C. C. GONZAGA, *Convergence of interior point algorithms for the monotone linear complementarity problem*, Math. Oper. Res., 21 (1996), pp. 1–25.
- [2] J. F. BONNANS AND F. A. POTRA, *Unfeasible path following algorithms for linear complementarity problems*, Math Oper. Res., to appear.
- [3] A. CHARNES, W. W. COOPER, AND R. M. THRALL, *A structure for classifying and characterizing efficiency and inefficiency in data envelopment analysis*, J. Productivity Anal., 2 (1991), pp. 197–237.
- [4] J. E. DENNIS AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice–Hall, Englewood Cliffs, NJ, 1983.
- [5] A. EL-BAKRY, R. A. TAPIA, T. TSUCHIYA, AND Y. ZHANG, *On the formulation of the primal-dual Newton interior-point method for nonlinear programming*, J. Optim. Theory Appl., 89 (1996), pp. 507–542.
- [6] A. EL-BAKRY, R. A. TAPIA, AND Y. ZHANG, *A study of indicators for identifying zero variables in interior-point methods*, SIAM Rev., 36 (1994), pp. 45–72.
- [7] C. C. GONZAGA, *Path-following methods for linear programming*, SIAM Rev., 34 (1992), pp. 167–224.
- [8] C. C. GONZAGA AND R. A. TAPIA, *On the convergence of the Mizuno–Todd–Ye algorithm to the analytic center of the solution set*, SIAM J. Optim., 7 (1997), pp. 47–65.
- [9] M. D. GONZÁLEZ-LIMA, *Effective Computation of the Analytic Center of the Solution Set in Linear Programming Using Primal-Dual Interior-Point Methods*, Technical report TR94-48, Rice University, Dept. of Computational and Applied Mathematics, Houston, TX, 1994.

- [10] O. GÜLER AND Y. YE, *Convergence behavior of some interior-point algorithms*, Math. Programming, 60 (1993), pp. 215–228.
- [11] M. KOJIMA, S. MIZUNO, AND A. YOSHISE, *A primal-dual interior point method for linear programming*, in Progress in Mathematical Programming, Interior-Point and Related Methods, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 29–47.
- [12] I. LUSTIG, R. MARSTEN, AND D. SHANNO, *On implementing Mehrotra's predictor-corrector interior-point method for linear programming*, SIAM J. Optim., 2 (1992), pp. 435–449.
- [13] L. MCLINDEN, *An analogue of Moreau's approximation theorem, with application to the non-linear complementarity problem*, Pacific J. Math., 88 (1980), pp. 101–161.
- [14] K. MCSHANE, *A superlinearly convergent $O(\sqrt{n}L)$ -iteration interior point algorithms for linear programming and the monotone linear complementarity problem*, SIAM J. Optim., 4 (1994), pp. 247–261.
- [15] N. MEGIDDO, *Pathways to the optimal set in linear programming*, in Progress in Mathematical Programming, Interior-Point and Related Methods, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 131–158.
- [16] S. MIZUNO, M. J. TODD, AND Y. YE, *On adaptive-step primal-dual interior-point algorithms for linear programming*, Math. Oper. Res., 18 (1993), pp. 964–981.
- [17] G. SONNEVEND, *"An analytical centre" for polyhedrons and new classes for linear (smooth, convex) programming*, in System Modelling and Optimization (Budapest, 1985), Lecture Notes in Control and Inform. Sci. 84, A. Prekopa, J. Szelezsan, and B. Strazicky, eds., Springer-Verlag, New York, 1985, pp. 866–876.
- [18] R. A. TAPIA, Y. ZHANG, AND Y. YE, *On the convergence of the iteration sequence in primal-dual interior-point methods*, Math. Programming, 68 (1995), pp. 141–154.
- [19] R. THOMPSON, P. S. DHARMAPALA, AND R. M. THRALL, *DEA sensitivity analysis of efficiency measures with an application to Kansas farming and Illinois coal mining*, in Data Envelopment Analysis: Theory, Methodology and Applications, A. Charnes, W. W. Cooper, A. Y. Lewin, and L. M. Seiford, eds., Kluwer, Boston, MA, 1994, pp. 293–422.
- [20] Y. YE, O. GÜLER, R. A. TAPIA, AND Y. ZHANG, *A quadratically convergent $O(\sqrt{n}L)$ -iteration algorithm for linear programming*, Math. Programming, 59 (1993), pp. 151–162.
- [21] Y. YE, R. A. TAPIA, AND Y. ZHANG, *A Superlinearly Convergent $O(\sqrt{n}L)$ -Iteration Algorithm for Linear Programming*, Technical report TR91-22, Dept. Mathematical Sciences, Rice University, Houston, TX, 1991.
- [22] Y. ZHANG AND A. EL-BAKRY, *A modified predictor-corrector algorithm for locating weighted centers in linear programming*, J. Optim. Theory Appl., 80 (1994), pp. 319–331.
- [23] Y. ZHANG AND R. A. TAPIA, *Superlinear and quadratic convergence of primal-dual interior point methods for linear programming revisited*, J. Optim. Theory Appl., 73 (1992), pp. 229–242.
- [24] Y. ZHANG AND R. A. TAPIA, *On the Convergence of Interior-Point Methods to the Center of Solution Set in Linear Programming*, Technical report TR91-30, Dept. Mathematical Sciences, Rice University, Houston, TX, 1991.
- [25] Y. ZHANG, R. A. TAPIA, AND J. E. DENNIS, *On the superlinear and quadratic convergence of primal-dual interior-point linear programming algorithms*, SIAM J. Optim., 2 (1992), pp. 304–324.
- [26] Y. ZHANG, R. A. TAPIA, AND F. A. POTRA, *On the superlinear convergence of interior-point algorithms for a general class of problems*, SIAM J. Optim., 3 (1993), pp. 413–422.

GLOBAL CONVERGENCE OF THE AFFINE SCALING ALGORITHM FOR CONVEX QUADRATIC PROGRAMMING*

RENATO D. C. MONTEIRO[†] AND TAKASHI TSUCHIYA[‡]

Abstract. In this paper we give a global convergence proof of the second-order affine scaling algorithm for convex quadratic programming problems, where the new iterate is the point which minimizes the objective function over the intersection of the feasible region with the ellipsoid centered at the current point and whose radius is a fixed fraction $\beta \in (0, 1]$ of the radius of the largest “scaled” ellipsoid inscribed in the nonnegative orthant. The analysis is based on the local Karmarkar potential function introduced by Tsuchiya. For any $\beta \in (0, 1)$ and without assuming any nondegeneracy assumption on the problem, it is shown that the sequences of primal iterates and dual estimates converge to optimal solutions of the quadratic program and its dual, respectively.

Key words. affine scaling algorithm, convex quadratic programming, interior point methods, global convergence, dual estimates, potential function

AMS subject classifications. 49M40, 65K05, 90C20, 90C25

PII. S1052623495283851

1. Introduction. The affine scaling (AS) algorithm for linear programming (LP) was introduced by Dikin [5] in 1967 but remained unknown to the western community until the late 1980’s. It was later rediscovered by Barnes [2] and Vanderbei, Meketon, and Freedman [27] as a natural simplification of Karmarkar’s algorithm [12]. Because of the theoretical and practical importance of the AS algorithm, there are a number of papers which study its global and local convergence [2, 5, 6, 7, 10, 19, 21, 22, 24, 25, 27, 26] and the behavior of its associated continuous trajectories [1, 3, 13, 14, 28]. As in the simplex algorithm, the analysis of the affine scaling algorithm for primal degenerate LP problems is much harder than for primal nondegenerate LP problems. By introducing a local Karmarkar potential function, Tsuchiya [21, 22] shows the global convergence of the short-step version of the AS algorithm where the next iterate minimizes the (linear) objective function over the intersection of the feasible region with the ellipsoid centered at the current point and whose radius is a fixed fraction β , $\beta \leq 1/8$, of the radius of the largest “scaled” ellipsoid inscribed in the nonnegative orthant. The global convergence of the long-step version of the affine scaling algorithm for degenerate LP problems, that is, the version where the next iterate is determined by taking a fixed fraction $\lambda \in (0, 1)$ of the whole step to the boundary of the feasible region, was established by Dikin [7] for any $\lambda \in (0, 1/2]$ and by Tsuchiya and Muramatsu [25] for any $\lambda \in (0, 2/3]$. We refer the reader to Monteiro, Tsuchiya, and Wang [15] and Saigal [16] for a simplified and self-contained analysis of the AS algorithm in the context of LP problems.

* Received by the editors March 29, 1995; accepted for publication (in revised form) November 2, 1996. The U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes. Copyright is owned by SIAM to the extent not limited by these rights.

<http://www.siam.org/journals/siopt/8-1/28385.html>

[†] School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332 (monteiro@isye.gatech.edu). The work of this author was based on research supported by the National Science Foundation under grants CCR-970048 and INT-9600343 and the Office of Naval Research under grants N00014-93-1-0234 and N00014-94-1-0340.

[‡] The Institute of Statistical Mathematics, 4-6-7 Minami-Azabu, Minato-Ku, Tokyo, 106, Japan (tsuchiya@sun312.ism.ac.jp). This author is supported in part by Grant-in-Aid for Encouragement of Young Scientists (A) 08780227 of the Ministry of Science, Education and Culture of Japan.

In this paper, we focus our attention on the second-order affine scaling algorithm for convex quadratic programming (QP), and prove its global convergence without any nondegeneracy assumption by using a local potential function similar to the one used in the LP context. This version of the AS algorithm was originally proposed by Dikin (see the references in his book [9]), and was later rediscovered by Ye [29] and Ye and Tse [31] after the introduction of Karmarkar’s algorithm [12]. The algorithm analyzed in this paper is similar to the short-step version of the AS algorithm for LP problems in that the next iterate is the point which minimizes the (quadratic) objective function over the intersection of the feasible region with the ellipsoid centered at the current point and whose radius is a fixed fraction β of the radius of the largest “scaled” ellipsoid inscribed in the nonnegative orthant. The papers [29, 31] assume that the QP problem is primal nondegenerate and also impose an extra dual nondegeneracy assumption involving the objective function. These assumptions considerably simplify the convergence analysis of the AS algorithm.

Global convergence of the second-order AS algorithm which drops the assumption of primal nondegeneracy but still keeps some sort of dual nondegeneracy is given in Tsuchiya [23] for strictly convex QP problems and any $\beta \in (0, 1/8]$. Sun [17] proves global convergence for this algorithm without imposing any nondegeneracy assumption. His analysis, which generalizes the one given in Tseng and Luo [19] for LP problems, is still restrictive in the sense that it allows only very small radius β , i.e., $\beta = 2^{-\mathcal{O}(L)}$, where L is the input size of the problem. In an unpublished manuscript, reported in the presentation [20], Tsuchiya proves global convergence of the second-order AS algorithm for strictly convex QP with $\beta \in (0, 1/8]$. Part of this paper is based on the ideas developed in that manuscript.

Convergence of the first-order AS algorithm for linearly constrained convex programming problems is proved in Gonzaga and Carlos [11] under the assumption of primal nondegeneracy. In this algorithm, the search direction is obtained by minimizing a first-order linear approximation of the objective function at the current point over the same ellipsoid mentioned above and the next point is determined by performing a line search along this direction.

Convergence analysis of the second-order AS algorithm for nonconvex QP problems under the assumption of primal and dual nondegeneracy can be found in the papers Ye [30] and Bonnans and Bouhtou [4].

This paper is organized as follows. In section 2 we describe the second-order AS algorithm for QP problems and review some basic results about this method. After introducing a local assumption which requires that a partition of the variables satisfy certain conditions, we prove in section 3 that the sequence of primal iterates and the sequence of dual estimates generated by the second-order AS algorithm with $\beta \in (0, 1)$ converge to optimal solutions of the given QP problem and its dual, respectively. Two important tools play a crucial role in the convergence analysis: the local potential function and an asymptotic decomposition formula of the scaled projection matrix near the boundary of the feasible region. Finally, we show in section 4 that the local assumption of section 3 is satisfied by some (in fact, unique) partition of variables. Appendix A presents a general framework for analyzing the asymptotic reduction of the local potential function. The asymptotic decomposition formula for the scaled projection matrix is presented in Tsuchiya [22] but its derivation is quite involved. For this reason we give a simplified proof of this formula in Appendix B.

In the first version of this paper, global convergence of the second-order AS algorithm is established for any fraction $\beta \in (0, 2/3]$, using arguments of the paper

[25] to derive an upper bound on the variation of the local potential function when $\beta \in (0, 2/3]$. Subsequently, Dikin and Roos [8] showed that an iteration of the short-step AS algorithm with $\beta = 1$ applied to a homogeneous linear program always reduces the associated Karmarkar potential function. Using their result, we show in the current version of the paper that the second-order AS algorithm for QP problems converges for any fraction $\beta \in (0, 1)$.

The following notation is used throughout our paper. We denote the vector of all ones by e . Its dimension is always clear from the context. $\mathbb{R}^p$, $\mathbb{R}_+^p$ and $\mathbb{R}_{++}^p$ denote the p -dimensional Euclidean space, the nonnegative orthant of $\mathbb{R}^p$ and the positive orthant of $\mathbb{R}^p$, respectively. The set of all $p \times q$ matrices with real entries is denoted by $\mathbb{R}^{p \times q}$. Given a vector $x \in \mathbb{R}^p$ and an index set $\alpha \subseteq \{1, \dots, p\}$, we denote the subvector $[x_i]_{i \in \alpha}$ by x_α . Given x and y in $\mathbb{R}^p$, $x \leq y$ means $x_i \leq y_i$ for every $i = 1, \dots, n$. If $\alpha \subseteq \{1, \dots, p\}$, $\beta \subseteq \{1, \dots, q\}$, and $Q \in \mathbb{R}^{p \times q}$, we let $Q_{\alpha\beta}$ denote the submatrix $[Q_{ij}]_{i \in \alpha, j \in \beta}$. If $\alpha = \{1, \dots, p\}$ we denote $Q_{\alpha\beta}$ simply by Q_β . For a vector w , we let $\chi[w]$ denote the largest component of w . The Euclidean norm, the 1-norm and the ∞ -norm are denoted by $\|\cdot\|$, $\|\cdot\|_1$ and $\|\cdot\|_\infty$, respectively. The diagonal matrix corresponding to a vector w is denoted by $\text{diag}(w)$, and the vector whose i th component is $1/w_i$ is denoted by w^{-1} . If J is a finite index set, then $|J|$ denotes its cardinality, that is the number of elements of J . We say that (B, N) is a partition of $\{1, \dots, p\}$ if $B \cup N = \{1, \dots, p\}$ and $B \cap N = \emptyset$. The superscript T denotes transpose.

2. Notation and preliminary results. In this section we describe the second-order AS algorithm for QP problems and review some basic results about this method.

We consider the following convex quadratic program (QP)

$$(1) \quad \begin{aligned} & \text{minimize} && f(x) = \frac{1}{2}x^T Qx + c^T x, \\ & \text{subject to} && Ax = b, \\ & && x \geq 0 \end{aligned}$$

and its associated dual QP

$$(2) \quad \begin{aligned} & \text{maximize} && b^T y - \frac{1}{2}x^T Qx, \\ & \text{subject to} && A^T y + s = \nabla f(x), \\ & && s \geq 0, \end{aligned}$$

where $c, x, s \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, and $Q \in \mathbb{R}^{n \times n}$ is a symmetric positive semidefinite matrix.

The following notation is used throughout the paper. The quantity $(x^T Qx)^{1/2}$ is simply denoted by $\|x\|_Q$. Clearly, $\|\cdot\|_Q$ is a norm only when Q is positive definite; in general, it fails to be a norm due to the fact that $\|x\|_Q = 0$ does not imply $x = 0$. If $\{\tau_k\}$ and $\{\beta_k\}$ are two sequences of real numbers and $\mathcal{K}$ is an infinite index set of nonnegative integers then the notation $\tau_k = \mathcal{O}(\beta_k)$ for $k \in \mathcal{K}$ means that there exists a scalar $r \geq 0$ such that $|\tau_k| \leq r\beta_k$ for every $k \in \mathcal{K}$ sufficiently large; clearly, if $\beta_k > 0$ for all $k \in \mathcal{K}$ then $|\tau_k| \leq r\beta_k$ holds for all $k \in \mathcal{K}$ by taking a larger scalar r if necessary. When $\mathcal{K}$ is equal to the set of all nonnegative integers, we simply write $\tau_k = \mathcal{O}(\beta_k)$. The notation $\tau_k \sim \beta_k$ for $k \in \mathcal{K}$ means that $\tau_k = \mathcal{O}(\beta_k)$ for $k \in \mathcal{K}$ and $\beta_k = \mathcal{O}(\tau_k)$ for $k \in \mathcal{K}$.

Given a partition (B, N) of $\{1, \dots, n\}$, we let

$$\mathcal{P}_N \equiv \{x \in \mathbb{R}^n \mid A_B x_B = b, x_N = 0\};$$

$$\begin{aligned}
 (3) \quad & \mathcal{P}_N^+ \equiv \{x \in \mathcal{P}_N \mid x_B \geq 0\}; \\
 & \mathcal{P}_N^{++} \equiv \{x \in \mathcal{P}_N \mid x_B > 0\}; \\
 & \mathcal{D}_B(x) \equiv \{(y, s) \in \mathbb{R}^m \times \mathbb{R}^n \mid A^T y + s = \nabla f(x), s_B = 0\}; \\
 & \mathcal{D}_B^+(x) \equiv \{(y, s) \in \mathcal{D}_B \mid s_N \geq 0\}; \\
 & \mathcal{D}_B^{++}(x) \equiv \{(y, s) \in \mathcal{D}_B \mid s_N > 0\}.
 \end{aligned}$$

When $N = \emptyset$, we denote the sets $\mathcal{P}_N$, $\mathcal{P}_N^+$ and $\mathcal{P}_N^{++}$ by $\mathcal{P}$, $\mathcal{P}^+$ and $\mathcal{P}^{++}$, respectively. The sets $\mathcal{P}^+$ and $\mathcal{P}^{++}$ are the sets of feasible solutions and strictly feasible solutions of problem (1). Similarly, when $B = \emptyset$, we denote the sets $\mathcal{D}_B(x)$, $\mathcal{D}_B^+(x)$, and $\mathcal{D}_B^{++}(x)$ by $\mathcal{D}(x)$, $\mathcal{D}^+(x)$, and $\mathcal{D}^{++}(x)$, respectively.

Throughout this paper, we make the following assumption.

ASSUMPTION 2.1. *The following assumptions are made with respect to (1):*

- (A) $\text{rank}(A) = m$;
- (B) $\mathcal{P}^{++} \neq \emptyset$;
- (C) *the set of optimal solutions of problem (1) is nonempty and bounded.*

We next describe the second-order affine scaling algorithm which is analyzed in this paper.

ALGORITHM AS. Let $x^0 \in \mathcal{P}^{++}$ and $\beta \in (0, 1)$ be given. Set $k = 0$.

step 1: compute the solution x^{k+1} of the problem

$$(4) \quad \min\{f(x) \mid Ax = b, \|(X^k)^{-1}(x - x^k)\|^2 \leq \beta^2\};$$

step 2: if $\|(X^k)^{-1}(x^{k+1} - x^k)\| < \beta$ then stop; otherwise, increment k by 1 and go to step 1.

end of Algorithm AS.

Using the fact that $\|(X^k)^{-1}(x^{k+1} - x^k)\| \leq \beta$, it follows that

$$(5) \quad (1 - \beta)x^k \leq x^{k+1} \leq (1 + \beta)x^k \quad \forall k \geq 0.$$

As a consequence, it follows that all iterates generated by Algorithm AS lie in $\mathcal{P}^{++}$.

PROPOSITION 2.2. *The following implications hold:*

- (a) *if $\|(X^k)^{-1}(x^{k+1} - x^k)\| < \beta$ for some k , then x^{k+1} is an optimal solution of problem (1);*
- (b) *if $\|(X^k)^{-1}(x^{k+1} - x^k)\| = \beta$ for all k , then every accumulation point of the sequence $\{x^k\}$ lies on the relative boundary of $\mathcal{P}^+$, that is, the set $\mathcal{P}^+ \setminus \mathcal{P}^{++}$.*

When case (a) of Proposition 2.2 occurs, Algorithm AS finds an optimal solution of problem (1) in a finite number of iterations. Throughout the paper we assume that case (b) of Proposition 2.2 holds. One of our goals will be to show that the sequence of iterates $\{x^k\}$ generated by Algorithm AS converges to an optimal solution of (1).

Throughout this paper, we let $X^k \equiv \text{diag}(x^k)$ and $g^k = \nabla f(x^k)$ for every k . The following quantities associated with the sequence $\{x^k\}$ generated by Algorithm AS play a fundamental role in the sequel:

$$(6) \quad y^k \equiv [A(X^k)^2 A^T]^{-1} A(X^k)^2 g^{k+1},$$

$$(7) \quad s^k \equiv g^{k+1} - A^T y^k = g^{k+1} - A^T [A(X^k)^2 A^T]^{-1} A(X^k)^2 g^{k+1},$$

$$(8) \quad P(x^k) \equiv I - X^k A^T [A(X^k)^2 A^T]^{-1} A X^k,$$

$$(9) \quad d^k \equiv (X^k)^2 s^k = X^k P(x^k) X^k g^{k+1}.$$

Using the fact that $\|(X^k)^{-1}(x^{k+1} - x^k)\| = \beta$ and x^{k+1} satisfies the optimality condition associated with (1), we can easily prove the following result.

LEMMA 2.3. *The following relations hold for every integer $k \geq 0$:*

$$(10) \quad x^{k+1} = x^k - \beta \frac{d^k}{\|(X^k)^{-1}d^k\|} = x^k - \beta \frac{(X^k)^2 s^k}{\|X^k s^k\|};$$

$$(11) \quad (g^{k+1})^T d^k = (s^k)^T d^k = \|(X^k)^{-1}d^k\|^2 = \|X^k s^k\|^2;$$

$$(12) \quad \beta \|X^k s^k\| = (g^{k+1})^T (x^k - x^{k+1}) \leq f(x^k) - f(x^{k+1}).$$

The following result is well known (see, for example, Ye [29] (or Ye and Tse [31]) and Sun [17]).

LEMMA 2.4. *The following statements hold:*

- (a) *the sequence $\{x^k\}$ converges to some point $x^* \in \mathcal{P}^+$;*
- (b) *$f(x^{k+1}) - f(x^*) \leq (1 - \beta/\sqrt{n})(f(x^k) - f(x^*))$ for every k sufficiently large;*
- (c) *there exists a constant $C_0 > 0$ such that $\|x^k - x^*\|^2 \leq C_0[f(x^k) - f^*]$ for every $k \geq 0$;*
- (d) *the sequence $\{(y^k, s^k)\}$ is bounded;*
- (e) *$\sum_{k=0}^{\infty} \|x^k s^k\| < \infty$ (hence, every accumulation point (y^*, s^*) of $\{(y^k, s^k)\}$ satisfies $x^* s^* = 0$).*

Proof. The proof of (a), (b), and (c) are given in Theorem 1 and Theorem 2 of Sun [17] under the conditions stated in Assumption 2.1. It has been proved in Vanderbei and Lagarias [26] that the matrix-valued mapping $U(x) \equiv (AX^2A^T)^{-1}AX^2$ is bounded on $\mathcal{P}^{++}$ whenever Assumption 2.1(A) holds. Since $y^k = U(x^k)\nabla f(x^{k+1})$, $s^k = \nabla f(x^{k+1}) - A^T y^k$ and $\{\nabla f(x^{k+1})\}$ is bounded, it follows that $\{(y^k, s^k)\}$ is bounded. Statement (e) is an immediate consequence of (12). $\square$

The following trivial result is frequently used in our presentation.

LEMMA 2.5. *If (y^*, s^*) is an accumulation point $\{(y^k, s^k)\}$ and (B, N) is a partition of $\{1, \dots, n\}$ such that $\{i \mid s_i^* \neq 0\} \subseteq N$ then $\nabla f(x^*)^T(x - x^*) = (s_N^*)^T x_N$ for any $x \in \mathcal{P}$.*

Proof. First note that $A^T y^k + s^k = \nabla f(x^{k+1})$ implies $A^T y^* + s^* = \nabla f(x^*)$. This relation together with the fact that $A(x - x^*) = 0$, $x^* s^* = 0$ and $s_B^* = 0$ imply that

$$\nabla f(x^*)^T(x - x^*) = (s^*)^T(x - x^*) = (s_N^*)^T x_N. \quad \square$$

3. Convergence to an optimal solution. In this section we review an important asymptotic decomposition result for the projection matrix $P(x^k)$. This result is used in several places of our presentation and is important in its own right. We also introduce a potential function that generalizes the one used in the context of linear programs and plays an important role in the convergence analysis of this section. After introducing a local assumption which guarantees the existence of a partition (B, N) of $\{1, \dots, n\}$ satisfying certain properties, we prove that the sequence of primal iterates and the sequence of dual estimates generated by the second-order AS algorithm with $\beta \in (0, 1)$ converge to optimal solutions of the given QP problem and its dual, respectively. The main goal of the next section is to show the existence of the partition (B, N) required for the analysis of this section.

We start this section by stating the following result on a decomposition of the projection matrix $P(x^k)$ whose proof can be found in Tsuchiya [22]. For completeness, a simpler proof of this result is given in the appendix. It is convenient to introduce the following notation. Let $P(x)$ denote the projection matrix onto the null space of the

matrix AX , $\tilde{P}_B(x_B)$ denote the projection matrix onto the null space of the matrix $A_B X_B$ and $\tilde{P}_N(x_N)$ denote the projection matrix onto the subspace $\{p_N \mid AXp = 0\} = \{p_N \mid A_N X_N p_N \in \text{Range}(A_B)\}$. Hence, for every $h \in \mathbb{R}^n$,

$$(13) \quad P(x)h = \operatorname{argmin}_p \{\|p - h\|^2 \mid AXp = 0\},$$

$$(14) \quad \tilde{P}_N(x_N)h_N = \operatorname{argmin}_{p_N} \{\|p_N - h_N\|^2 \mid AXp = 0\},$$

$$(15) \quad \tilde{P}_B(x_B)h_B = \operatorname{argmin}_{p_B} \{\|p_B - h_B\|^2 \mid A_B X_B p_B = 0\}.$$

LEMMA 3.1. *For every $x \in \mathcal{P}^{++}$, let*

$$(16) \quad R(x) \equiv P(x) - \begin{pmatrix} \tilde{P}_N(x_N) & 0 \\ 0 & \tilde{P}_B(x_B) \end{pmatrix}.$$

Then the following statements hold:

(a) *for every $x \in \mathcal{P}^{++}$, $R(x)$ is a symmetric matrix such that*

$$(17) \quad R_{NN}(x) = \tilde{P}_N(x_N)R_{NN}(x)\tilde{P}_N(x_N), \quad R_{NB}(x) = \tilde{P}_N(x_N)R_{NB}(x);$$

(b) *there exists a constant $C_1 > 0$ such that for every $x \in \mathcal{P}^{++}$ with $\Delta_N(x) \equiv \|x_N\| \|x_B^{-1}\|$ sufficiently small, there hold*

$$\begin{aligned} \|R_{NN}(x)\| &\leq C_1 \Delta_N(x)^2, \\ \|R_{BB}(x)\| &\leq C_1 \Delta_N(x)^2, \\ \|R_{BN}(x)\| &= \|R_{NB}^T(x)\| \leq C_1 \Delta_N(x). \end{aligned}$$

Throughout this section, we make a local assumption that will be used to show that Algorithm AS converges to an optimal solution of problem (1) (see Theorem 3.6). This assumption imposes conditions on the way the sequences $\{x^k\}$ and $\{(y^k, s^k)\}$ generated by Algorithm AS behave. In section 4, we show that it holds without assuming any nondegeneracy assumption on problem (1).

Before stating the local assumption, we introduce the following notation which is used throughout our presentation.

$$(18) \quad \begin{aligned} g^* &= \nabla f(x^*), \\ \tilde{s}^k &= g^* - A^T[A(X^k)^2 A^T]^{-1} A(X^k)^2 g^*, \\ \tilde{d}^k &= X^k P(x^k) X^k g^* = (X^k)^2 \tilde{s}^k, \\ N_* &= \{i \mid x_i^* = 0\}, \quad B_* = \{i \mid x_i^* > 0\}. \end{aligned}$$

Note that $\tilde{d}^k$ and $\tilde{s}^k$ are the search direction and the dual estimate corresponding to the affine scaling algorithm applied to the LP problem whose objective function is $(g^*)^T x = (\lim_{k \rightarrow \infty} \nabla f(x^k))^T x$.

LOCAL ASSUMPTION 3.2. *There exists a partition (B, N) of $\{1, \dots, n\}$ such that $N \subseteq N_*$ and the following properties are satisfied:*

- (A) *the sequence $\{\Delta_N^k\}$ defined as $\Delta_N^k \equiv \|x_N^k\| \|(x_B^k)^{-1}\|$ for all $k \geq 0$ satisfies $\sum_{k=0}^{\infty} \Delta_N^k < \infty$;*
- (B) *there exists $\tau_0 > 0$ such that $(g^*)^T(x^k - x^*) \geq \tau_0 \|x_N^k\|$ for every k sufficiently large;*
- (C) *the sequences $\{u^k\}$ and $\{\tilde{u}^k\}$ defined as*

$$(19) \quad u^k \equiv \frac{(X^k)^{-1} \tilde{d}^k}{(g^*)^T(x^k - x^*)}, \quad \tilde{u}^k \equiv \frac{(X^k)^{-1} \tilde{d}^k}{(g^*)^T(x^k - x^*)} \quad \forall k \geq 0,$$

satisfy the following properties: $\{u^k - \tilde{u}^k\}$ is bounded and $u_N^k - \tilde{u}_N^k = \mathcal{O}(\Delta_N^k)$;
(D) $\lim_{k \rightarrow \infty} s_i^k = 0$ for every $i \in N_* \setminus N$.

Our goal in the remaining of this section is to show that Local Assumption 3.2 is a sufficient condition for $\{x^k\}$ and $\{(y^k, s^k)\}$ to converge to optimal solutions of (1) and (2), respectively. We start with the following lemma.

LEMMA 3.3. *The following properties hold:*

- (a) $\{\tilde{u}^k\}$ is bounded;
- (b) $|e^T \tilde{u}_N^k - 1| = \mathcal{O}((\Delta_N^k)^2)$ (and hence $\|\tilde{u}_N^k\| \geq 1/\sqrt{|N|} - \mathcal{O}((\Delta_N^k)^2)$);
- (c) $\|\tilde{u}_B^k\| = \mathcal{O}(\Delta_N^k)$.

Proof. First note that

$$(20) \quad (X^k)^{-1} \tilde{d}^k = P(x^k) X^k g^* = P(x^k) X^k \bar{s},$$

for any $(\bar{y}, \bar{s}) \in \mathcal{D}(x^*)$. Let $(\bar{y}, \bar{s})$ be an accumulation point of the sequence $\{(y^k, s^k)\}$. Clearly, $(\bar{y}, \bar{s}) \in \mathcal{D}(x^*)$ and, by Local Assumption 3.2(D) and Lemma 2.4(e), we have $\bar{s}_B = 0$. Then, by (19), (20), and Local Assumption 3.2(B), we obtain

$$\|\tilde{u}^k\| = \frac{\|P(x^k) X^k \bar{s}\|}{(g^*)^T(x^k - x^*)} \leq \frac{\|X^k \bar{s}\|}{(g^*)^T(x^k - x^*)} \leq \frac{\|\bar{s}_N\| \|x_N^k\|}{(g^*)^T(x^k - x^*)} \leq \mathcal{O}(1),$$

and hence (a) follows. By (20), Lemma 3.1 and the fact that $\bar{s}_B = 0$, we obtain

$$(21) \quad (X^k)^{-1} \tilde{d}^k = P(x^k) X^k \bar{s} = \begin{pmatrix} \left\{ \tilde{P}_N(x_N^k) + R_{NN}(x^k) \right\} X_N^k \bar{s}_N \\ R_{BN}(x^k) X_N^k \bar{s}_N \end{pmatrix},$$

where $\|R_{BN}(x^k)\| = \mathcal{O}(\Delta_N^k)$ and $\|R_{NN}(x^k)\| = \mathcal{O}((\Delta_N^k)^2)$. Hence,

$$\|\tilde{u}_B^k\| = \frac{\|R_{BN}(x^k) X_N^k \bar{s}_N\|}{(g^*)^T(x^k - x^*)} \leq \frac{\mathcal{O}(\|x_N^k\| \Delta_N^k)}{(g^*)^T(x^k - x^*)} \leq \mathcal{O}(\Delta_N^k),$$

where the last inequality follows from Local Assumption 3.2(B). This shows (c). To show (b), we claim that

$$(22) \quad e^T \tilde{P}_N(x_N^k) X_N^k \bar{s}_N = (g^*)^T(x^k - x^*), \quad \forall k \geq 0.$$

Before showing (22), we show how (22) implies (b). Indeed, by (19), (21), and (22), we obtain

$$\begin{aligned} |e^T \tilde{u}_N^k - 1| &= \left| \frac{e^T \left[(X_N^k)^{-1} \tilde{d}_N^k - \tilde{P}_N(x_N^k) X_N^k \bar{s}_N \right]}{(g^*)^T(x^k - x^*)} \right| \\ &= \left| \frac{e^T R_{NN}(x^k) X_N^k \bar{s}_N}{(g^*)^T(x^k - x^*)} \right| \leq \mathcal{O}((\Delta_N^k)^2), \end{aligned}$$

where the last inequality follows from Local Assumption 3.2(B) and the fact that $\|R_{NN}(x^k)\| = \mathcal{O}((\Delta_N^k)^2)$. We have thus shown (b). It remains to verify (22). Let $p_N^k \equiv \tilde{P}_N(x_N^k) X_N^k \bar{s}_N$. Since p_N^k solves problem (14) with $x_N = x_N^k$ and $h_N = X_N^k \bar{s}_N$, the optimality condition for this problem yields

$$(23) \quad (A_B)^T \dot{y}^k = 0, \quad X_N^k (A_N)^T \dot{y}^k = p_N^k - X_N^k \bar{s}_N,$$

for some vector $\hat{y}^k \in \mathfrak{R}^m$. Since $b = A_{B_*} x_{B_*}^* \in \text{Range}(A_B)$, it follows that $A_N x_N^k = b - A_B x_B^k \in \text{Range}(A_B)$. Using this observation and the first equality in (23), we obtain $e^T [X_N^k (A_N)^T \hat{y}^k] = (A_N x_N^k)^T \hat{y}^k = 0$. Hence, the second equality in (23) yields

$$e^T p_N^k = \bar{s}_N^T x_N^k = (g^*)^T (x^k - x^*),$$

where the last equality follows from Lemma 2.5. This shows the claim. $\square$

The following potential function, usually referred to as the local Karmarkar potential function, plays an important role in our analysis. It was introduced by Tsuchiya to analyze the affine scaling algorithms for LP in [22] and for QP in [23]. For every $x \in \mathcal{P}_{++}$ satisfying $(g^*)^T (x - x^*) > 0$, let

$$(24) \quad \psi_N(x) = |N| \log[(g^*)^T (x - x^*)] - \sum_{i \in N} \log x_i.$$

The following result provides an upper bound for the difference $\psi_N(x^{k+1}) - \psi_N(x^k)$.

LEMMA 3.4. *Suppose that Local Assumption 3.2 holds. Then,*

$$(25) \quad \psi_N(x^{k+1}) - \psi_N(x^k) \leq \mathcal{O}(\Delta_N^k).$$

Moreover, if $\lim_{k \rightarrow \infty} [\psi_N(x^{k+1}) - \psi_N(x^k)] = 0$ then $\lim_{k \rightarrow \infty} \tilde{u}_N^k = \lim_{k \rightarrow \infty} u_N^k = e/|N|$.

Proof. Let (y^*, s^*) be an accumulation point of $\{(y^k, s^k)\}$. Observe that Local Assumption 3.2(D) implies that $\{i \mid s_i^* \neq 0\} \subseteq N$. Hence, it follows from Lemma 2.5 that $(g^*)^T (x^k - x^*) = (s_N^*)^T x_N^k$ for every $k \geq 0$. Using this observation, Local Assumption 3.2(B) and relation (5), we obtain

$$(26) \quad \frac{(g^*)^T (x^{k+1} - x^*)}{(g^*)^T (x^k - x^*)} \geq \frac{\tau_0 \|x_N^{k+1}\|}{(s_N^*)^T x_N^k} \geq \frac{\tau_0 \|x_N^{k+1}\|}{\|s_N^*\| \|x_N^k\|} \geq \frac{\tau_0 (1 - \beta)}{\|s_N^*\|}.$$

Using relation (10), the fact that $(g^*)^T \tilde{d}^k = \|(X^k)^{-1} \tilde{d}^k\|^2$ for all k , the definition of u^k and $\tilde{u}^k$, Lemma 3.3(b) and conditions (B) and (C) of Local Assumption 3.2, we obtain

$$\begin{aligned} & \frac{(g^*)^T (x^{k+1} - x^*)}{(g^*)^T (x^k - x^*)} \\ &= 1 - \frac{\beta}{(g^*)^T (x^k - x^*)} \frac{(g^*)^T d^k}{\|(X^k)^{-1} d^k\|} \\ &= 1 - \frac{\beta}{(g^*)^T (x^k - x^*)} \frac{(g^*)^T \tilde{d}^k}{\|(X^k)^{-1} \tilde{d}^k\|} + \frac{\beta}{(g^*)^T (x^k - x^*)} \frac{(g^*)^T (\tilde{d}^k - d^k)}{\|(X^k)^{-1} \tilde{d}^k\|} \\ &= 1 - \frac{\beta}{(g^*)^T (x^k - x^*)} \frac{\|(X^k)^{-1} \tilde{d}^k\|^2}{\|(X^k)^{-1} \tilde{d}^k\|} + \frac{\beta}{(g^*)^T (x^k - x^*)} \frac{(s_N^*)^T (\tilde{d}_N^k - d_N^k)}{\|(X^k)^{-1} \tilde{d}^k\|} \\ &= 1 - \beta \frac{\|\tilde{u}^k\|^2}{\|u^k\|} + \frac{\beta}{(g^*)^T (x^k - x^*)} \frac{(X_N^k s_N^*)^T (\tilde{u}_N^k - u_N^k)}{\|u^k\|} \\ &= 1 - \lambda_k \|\tilde{u}^k\| + \mathcal{O} \left(\frac{\|x_N^k\|}{(g^*)^T (x^k - x^*)} \frac{\|\tilde{u}_N^k - u_N^k\|}{\|u_N^k\|} \right) \\ (27) \quad &= 1 - \lambda_k \|\tilde{u}^k\| + \mathcal{O}(\Delta_N^k), \end{aligned}$$

where $\lambda_k \equiv \beta \|\tilde{u}^k\|/\|u^k\|$. Using relations (5) and (10), the definition of λ_k , u^k and $\tilde{u}^k$, Lemma 3.3(b), and Local Assumption 3.2(C), we obtain for every $i \in N$ that

$$\begin{aligned}
 1 - \beta &\leq \frac{x_i^{k+1}}{x_i^k} = 1 - \beta \frac{(x_i^k)^{-1} d_i^k}{\|(X^k)^{-1} d^k\|} \\
 &= 1 - \beta \frac{(x_i^k)^{-1} \tilde{d}_i^k}{\|(X^k)^{-1} d^k\|} + \beta \frac{(x_i^k)^{-1} (\tilde{d}_i^k - d_i^k)}{\|(X^k)^{-1} d^k\|} \\
 &= 1 - \beta \frac{\tilde{u}_i^k}{\|u^k\|} + \beta \frac{\tilde{u}_i^k - u_i^k}{\|u^k\|} \\
 &= 1 - \lambda_k \frac{\tilde{u}_i^k}{\|\tilde{u}^k\|} + \mathcal{O}\left(\frac{\|\tilde{u}_N^k - u_N^k\|}{\|u^k\|}\right) \\
 (28) \quad &= 1 - \lambda_k \frac{\tilde{u}_i^k}{\|\tilde{u}^k\|} + \mathcal{O}(\Delta_N^k).
 \end{aligned}$$

Relations (26), (27), and (28), Lemma A.1, Local Assumption 3.2(A) and the expression

$$\psi_N(x^{k+1}) - \psi_N(x^k) = |N| \log \left(\frac{(g^*)^T(x^{k+1} - x^*)}{(g^*)^T(x^k - x^*)} \right) - \sum_{i \in N} \log \left(\frac{x_i^{k+1}}{x_i^k} \right)$$

then imply that

$$(29) \quad \psi_N(x^{k+1}) - \psi_N(x^k) = \phi_N(\tilde{u}^k, \lambda_k) + \mathcal{O}(\Delta_N^k),$$

where

$$\phi_N(\tilde{u}^k, \lambda_k) \equiv |N| \log(1 - \lambda_k \|\tilde{u}^k\|) - \sum_{i \in N} \log \left(1 - \lambda_k \frac{\tilde{u}_i^k}{\|\tilde{u}^k\|} \right).$$

We will now show that $\{\tilde{u}^k\}$ and $\{\lambda_k\}$ satisfy all the conditions (a)–(e) of Proposition A.5. Indeed, conditions (b), (c), and (d) follow immediately from Lemma 3.3 and the fact that $\lim_{k \rightarrow \infty} \Delta_N^k = 0$, in view of Local Assumption 3.2(A). Condition (a) follows from (26), (27) and the fact that $\lim_{k \rightarrow \infty} \Delta_N^k = 0$. It remains to verify that (e) holds. Due to Local Assumption 3.2(C), the fact that $\lim_{k \rightarrow \infty} \Delta_N^k = 0$ and Lemma 3.3, we have $\lim_{k \rightarrow \infty} \|\tilde{u}_B^k\| = 0$, $\lim_{k \rightarrow \infty} \|u_N^k\|/\|\tilde{u}_N^k\| = 1$ and $\liminf_{k \rightarrow \infty} \|u_N^k\| \geq 1/\sqrt{|N|}$. Using these conclusions, we obtain

$$\limsup_{k \rightarrow \infty} \lambda_k = \beta \limsup_{k \rightarrow \infty} \frac{\|\tilde{u}^k\|}{\|u^k\|} \leq \beta \limsup_{k \rightarrow \infty} \frac{\|\tilde{u}_N^k\| + \|\tilde{u}_B^k\|}{\|u_N^k\|} = \beta < 1$$

and

$$\liminf_{k \rightarrow \infty} \lambda_k = \beta \liminf_{k \rightarrow \infty} \frac{\|\tilde{u}^k\|}{\|u^k\|} \geq \beta \frac{\liminf_{k \rightarrow \infty} \|\tilde{u}_N^k\|}{\sup_k \|u^k\|} > 0,$$

from which (e) of Proposition A.5 follows.

Relation (25) now follows from inequality (103) of Proposition A.5, Lemma 3.3 and (29). Now, if $\lim_{k \rightarrow \infty} [\psi_N(x^{k+1}) - \psi_N(x^k)] = 0$ then, by (29) and the fact that $\lim_{k \rightarrow \infty} \Delta_N^k = 0$, we have $\lim_{k \rightarrow \infty} \phi_N(\tilde{u}^k, \lambda_k) = 0$ and hence, by Proposition A.5 and Local Assumption 3.2(C), we obtain $\lim_{k \rightarrow \infty} \tilde{u}_N^k = \lim_{k \rightarrow \infty} u_N^k = e/|N|$. $\square$

The next result shows that $\lim_{k \rightarrow \infty} u_N^k = e/|N|$.

LEMMA 3.5. *Suppose that Local Assumption 3.2 holds. Then,*

$$(30) \quad \lim_{k \rightarrow \infty} u_N^k = \frac{1}{|N|} e.$$

Proof. First observe that the sequence $\{\psi_N(x^k)\}$ is bounded below since

$$(31) \quad \begin{aligned} \psi_N(x^k) &= |N| \log \left(\frac{(g^*)^T(x^k - x^*)}{\sum_{i \in N} x_i^k} \right) + |N| \log \left(\sum_{i \in N} x_i^k \right) - \sum_{i \in N} \log x_i^k \\ &\geq |N| \log \left(\frac{\tau_0}{|N|^{1/2}} \right) + |N| \log |N|, \end{aligned}$$

where the last inequality is due to the inequalities $\|x_N^k\| \geq (\sum_{i \in N} x_i^k)/\sqrt{|N|}$ and $(\sum_{i \in N} x_i^k)/|N| \geq (\prod_{i \in N} x_i^k)^{1/|N|}$ and Local Assumption 3.2(B). Assume for contradiction that (30) does not hold. In view of Lemma 3.4 and the fact that $\lim_{k \rightarrow \infty} \Delta_N^k = 0$, this implies that $\liminf_{k \rightarrow \infty} [\psi_N(x^{k+1}) - \psi_N(x^k)] < 0$. Hence, there exist $\gamma > 0$ and an infinite index set $\mathcal{K}$ such that $\psi_N(x^{k+1}) - \psi_N(x^k) \leq -\gamma$ for all $k \in \mathcal{K}$. This implies

$$(32) \quad \sum_{k \in \mathcal{K}} [\psi_N(x^{k+1}) - \psi_N(x^k)] = -\infty.$$

Moreover, by Lemma 3.4, there exists a constant $L_0 > 0$ such that $\psi_N(x^{k+1}) - \psi_N(x^k) \leq L_0 \Delta_N^k$ for all $k \geq 0$. Hence,

$$\sum_{k \notin \mathcal{K}} [\psi_N(x^{k+1}) - \psi_N(x^k)] \leq L_0 \sum_{k \notin \mathcal{K}} \Delta_N^k < \infty,$$

where the last inequality is due to Local Assumption 3.2(A). Combining this last relation with (32), we conclude that

$$\sum_{k=0}^{\infty} [\psi_N(x^{k+1}) - \psi_N(x^k)] = -\infty,$$

and hence, $\lim_{k \rightarrow \infty} \psi_N(x^k) = -\infty$. Since this contradicts (31), the result follows. $\square$

The next result establishes that the sequence $\{x^k\}$ generated by Algorithm AS converges to an optimal solution of problem (1) and that the sequence of dual estimates $\{(y^k, s^k)\}$ converges to an optimal solution of (2).

THEOREM 3.6. *Suppose that the partition (B, N) satisfies Local Assumption 3.2. Then the limit point x^* of the sequence $\{x^k\}$ generated by Algorithm AS is an optimal solution of problem (1) and the associated sequence of dual estimates $\{(y^k, s^k)\}$ converges to an optimal solution (y^*, s^*) of the dual problem (2) having the following property: (y^*, s^*) is the unique optimal solution of the problem*

$$(33) \quad \begin{aligned} &\text{maximize} && \sum_{i \in N} \log s_i, \\ &\text{subject to} && A^T y + s = g^*, \\ &&& s_B = 0, \quad s_N > 0. \end{aligned}$$

Proof. Let (y^*, s^*) denote the optimal solution of (33). Since A has full row rank, it follows that if $\lim_{k \rightarrow \infty} s^k = s^*$ then $\lim_{k \rightarrow \infty} y^k = y^*$. Hence, it is sufficient to show that $\lim_{k \rightarrow \infty} s^k = s^*$. It can be easily verified that s^* is the unique point satisfying the following relations:

$$\begin{aligned} (34) \quad & s_N > 0, \quad s_B = 0; \\ (35) \quad & s \in g^* + \text{Range}(A^T); \\ (36) \quad & A_N(s_N)^{-1} \in \text{Range}(A_B), \end{aligned}$$

where $(s_N)^{-1}$ denotes the subvector $[(s_i)^{-1}]_{i \in N}$. We will show that every accumulation point of the (bounded) sequence $\{s^k\}$ satisfies relations (34), (35), and (36), thereby showing that $\lim_{k \rightarrow \infty} s^k = s^*$. Indeed, let s^∞ denote an accumulation point of $\{s^k\}$, that is, $s^\infty = \lim_{k \in \mathcal{K}} s^k$ for some infinite index set $\mathcal{K}$. By Lemma 2.4(e) and Local Assumption 3.2(D), we know that $s_B^\infty = 0$. Letting k tend to ∞ in the relation $s^k \in g^{k+1} + \text{Range}(A^T)$, we conclude that $s^\infty \in g^* + \text{Range}(A^T)$. By Lemma 3.5, we have

$$\lim_{k \rightarrow \infty} u_N^k = \lim_{k \rightarrow \infty} \frac{X_N^k s_N^k}{(g^*)^T(x^k - x^*)} = \frac{1}{|N|} e,$$

where $X_N^k = \text{diag}(x_N^k)$, and by Local Assumption 3.2(B), we know that the quantity $\|x_N^k\|/[(g^*)^T(x^k - x^*)]$ is bounded. From these two observations, we can easily see that $s_N^\infty > 0$ and that

$$(37) \quad \lim_{k \in \mathcal{K}} \frac{|N|}{(g^*)^T(x^k - x^*)} x_N^k = (s_N^\infty)^{-1}.$$

Since $b \in \text{Range}(A_{B_*}) \subseteq \text{Range}(A_B)$ and $Ax^k = b$, we conclude that

$$A_N \left(\frac{|N|}{(g^*)^T(x^k - x^*)} x_N^k \right) \in \text{Range}(A_B), \quad \forall k \geq 0.$$

Letting $k \in \mathcal{K}$ tend to ∞ in this relation and using (37), we obtain that

$$A_N(s_N^\infty)^{-1} \in \text{Range}(A_B).$$

We have thus shown that s^∞ satisfies (34), (35) and (36), and hence it follows that $s^\infty = s^*$. $\square$

4. Global convergence proof. The main goal of this section is to show the existence of the partition (B, N) satisfying the conditions (A)–(D) of Assumption 3.2.

The following technical lemma is easy to verify and is frequently used in our presentation.

LEMMA 4.1. *Let $F \in \mathbb{R}^{p \times q}$ be given. Then, there exists a constant $C_2 = C_2(F)$ with the following property: for any $f \in \mathbb{R}^p$ such that the system $Fw = f$ is feasible and any $z \in \mathbb{R}^q$, there exists a solution $\bar{w}$ of $Fw = f$ such that*

$$\|\bar{w} - z\| \leq C_2 \|f - Fz\|.$$

Our next goal is to show that for index set N such that $N \subseteq N_*$, we have $f(x^k) - f(x^*) = \mathcal{O}(\|x_N^k\|)$ (see Theorem 4.4). This result plays a fundamental role in the analysis of this section.

We use the following notation. Given a point $x \in \mathcal{P}^{++}$, let

$$(38) \quad \mathcal{E}(x) = \mathcal{E}(x; \beta) \equiv \{w \in \mathcal{P} \mid \|X^{-1}(w - x)\| \leq \beta\}.$$

LEMMA 4.2. $f(x^{k+1}) - f(x^*) = \mathcal{O}(\|X^k s^k\|)$.

Proof. Consider the point $\tilde{x}^{k+1}$ defined as

$$\tilde{x}^{k+1} \equiv x^k + \frac{\beta(x^* - x^k)}{\|(X^k)^{-1}(x^* - x^k)\|}.$$

Clearly, $\tilde{x}^{k+1} \in \mathcal{E}(x^k)$ (see (38)). Since x^{k+1} is the optimal solution of (4), we know that $(g^{k+1})^T(x - x^{k+1}) \geq 0$ for all $x \in \mathcal{E}(x^k)$. In particular, $x = \tilde{x}^{k+1}$ yields

$$(g^{k+1})^T \left(x^k - x^{k+1} + \frac{\beta(x^* - x^k)}{\|(X^k)^{-1}(x^* - x^k)\|} \right) \geq 0,$$

or equivalently,

$$(39) \quad \|(X^k)^{-1}(x^* - x^k)\| - \beta \geq \frac{\beta(g^{k+1})^T(x^{k+1} - x^*)}{(g^{k+1})^T(x^k - x^{k+1})}.$$

It is easy to see that $\lim_{k \rightarrow \infty} \|(X^k)^{-1}(x^* - x^k)\| = |N_*|^{1/2}$. This relation together with (12), (39), and the inequality $f(x^{k+1}) - f(x^*) \leq (g^{k+1})^T(x^{k+1} - x^*)$ imply

$$\begin{aligned} \limsup_{k \rightarrow \infty} \frac{f(x^{k+1}) - f(x^*)}{\|X^k s^k\|} &\leq \beta \limsup_{k \rightarrow \infty} \frac{(g^{k+1})^T(x^{k+1} - x^*)}{(g^{k+1})^T(x^k - x^{k+1})} \\ &\leq \limsup_{k \rightarrow \infty} (\|(X^k)^{-1}(x^* - x^k)\| - \beta) = |N_*|^{1/2} - \beta, \end{aligned}$$

and the result follows. $\square$

LEMMA 4.3. Assume that $\{a_k\}$ is an unbounded sequence of positive scalars such that $a_{k+1} = \mathcal{O}(a_k)$. Then there exists a subsequence $\{a_k\}_{k \in \mathcal{K}}$ such that $\lim_{k \in \mathcal{K}} a_k = \infty$ and $a_k \leq a_{k+1}$ for all $k \in \mathcal{K}$.

Proof. Let $\mathcal{K}_0 \equiv \{k \mid a_{k-1} \leq a_k\}$. We first show that the subsequence $\{a_k\}_{k \in \mathcal{K}_0}$ is unbounded. Indeed, assume for contradiction that, for some $L_0 > 0$, $a_k \leq L_0$ for every $k \in \mathcal{K}_0$. Let k_0 denote the smallest index in $\mathcal{K}_0$. We claim that $a_k \leq L_0$ for every $k \geq k_0$, a fact that contradicts the assumption that $\{a_k\}$ is unbounded. To prove the claim, let $k \geq k_0$ be given. Clearly, if $k \in \mathcal{K}_0$ then $a_k \leq L_0$. If $k \notin \mathcal{K}_0$ then let $j(k)$ denote the largest index $j \in \mathcal{K}_0$ such that $j < k$. Since $j(k) \in \mathcal{K}_0$ and $j(k) + 1, \dots, k \notin \mathcal{K}_0$, we have $L_0 \geq a_{j(k)} > a_{j(k)+1} > \dots > a_k$. This shows that $a_k \leq L_0$ for every $k \geq k_0$ and, hence, that $\{a_k\}_{k \in \mathcal{K}_0}$ is unbounded. Hence, there exists an infinite index set $\mathcal{K}$ such that $a_k \leq a_{k+1}$ for every $k \in \mathcal{K}$ and $\lim_{k \in \mathcal{K}} a_{k+1} = \infty$. Since $a_{k+1} = \mathcal{O}(a_k)$, it also follows that $\lim_{k \in \mathcal{K}} a_k = \infty$. Hence, the result follows. $\square$

Now we are ready to see the theorem.

THEOREM 4.4. Let $N \subseteq N_*$ be given. Then the following statements hold:

- (a) $f(x^k) - f(x^*) = \mathcal{O}(\|x_N^k\|)$;
- (b) $\|x^k - x^*\|^2 = \mathcal{O}(\|x_N^k\|)$.

Proof. Observe that (b) follows as a consequence of (a) and Lemma 2.4(c). We next show (a). Assume for contradiction that $a_k \equiv [f(x^k) - f(x^*)]/\|x_N^k\|$ is unbounded. Clearly, we have $a_{k+1} \leq (1 - \beta)^{-1}a_k = \mathcal{O}(a_k)$. Hence, it follows from Lemma 4.3 that there exists an infinite index set $\mathcal{K}$ such that $\lim_{k \in \mathcal{K}} a_k = \infty$ and

$a_k \leq a_{k+1}$ for all $k \in \mathcal{K}$. Using the definition of a_k , the last inequality and Lemma 2.4(b), we obtain

$$(40) \quad \frac{\|x_N^{k+1}\|}{\|x_N^k\|} \leq \frac{f(x^{k+1}) - f(x^*)}{f(x^k) - f(x^*)} \leq \gamma_0 < 1 \quad \forall k \in \mathcal{K},$$

where $\gamma_0 = 1 - \beta/\sqrt{n} < 1$. It is easy to see that this relation implies that

$$(41) \quad \liminf_{k \in \mathcal{K}} \|(X_N^k)^{-1}(x_N^{k+1} - x_N^k)\| > 0.$$

We can easily verify the existence of an index set $\hat{N} \subseteq \{1, \dots, n\}$ and an infinite index set $\mathcal{K}_0 \subseteq \mathcal{K}$ such that

$$(42) \quad \|x_N^k\| = \mathcal{O}(\|x_N^k\|) \quad \forall k \in \mathcal{K}_0,$$

$$(43) \quad \lim_{k \in \mathcal{K}_0} \frac{x_i^k}{\|x_N^k\|} = \infty \quad \forall i \notin \hat{N}.$$

These relations imply that $\lim_{k \in \mathcal{K}} \Delta_{\hat{N}}^k = 0$. Clearly, $N \subseteq \hat{N} \subseteq N_*$. In view of (42) and the fact that $\lim_{k \in \mathcal{K}} a_k = \infty$, we obtain

$$(44) \quad \lim_{k \in \mathcal{K}_0} \frac{f(x^k) - f(x^*)}{\|x_N^k\|} = \infty.$$

Using Lemma 4.1 with $z = x^{k+1}$, it follows that for every $k \geq 0$ the feasible system (e.g., $x = x^k$ is a feasible solution)

$$(45) \quad Ax = b, \quad x_{\hat{N}} = x_{\hat{N}}^k$$

has a solution $\hat{x}^{k+1}$ such that

$$(46) \quad \|\hat{x}^{k+1} - x^{k+1}\| = \mathcal{O}(\|x_N^k - x_N^{k+1}\|) = \mathcal{O}(\|x_N^k\|).$$

We will now show that the half line starting at x^k and passing through the point $\hat{x}^{k+1}$ contains a point in $\mathcal{E}(x^k)$ whose f -function value is smaller than $f(x^{k+1})$ whenever $k \in \mathcal{K}_0$ is sufficiently large, a fact that contradicts the definition of x^{k+1} . We first show that

$$(47) \quad \limsup_{k \in \mathcal{K}_0} \|(X^k)^{-1}(\hat{x}^{k+1} - x^k)\| < \beta.$$

Indeed, the fact that $\hat{x}_{\hat{N}}^{k+1} = x_{\hat{N}}^k$ implies

$$(48) \quad \begin{aligned} & \|(X^k)^{-1}(\hat{x}^{k+1} - x^k)\|^2 \\ &= \|(X_{\hat{B}}^k)^{-1}(\hat{x}_{\hat{B}}^{k+1} - x_{\hat{B}}^k)\|^2 \\ &\leq \|(X_{\hat{B}}^k)^{-1}(x_{\hat{B}}^{k+1} - x_{\hat{B}}^k)\|^2 + \|(X_{\hat{B}}^k)^{-1}(\hat{x}_{\hat{B}}^{k+1} - x_{\hat{B}}^{k+1})\|^2 \\ &\quad + 2\|(X_{\hat{B}}^k)^{-1}(\hat{x}_{\hat{B}}^{k+1} - x_{\hat{B}}^{k+1})\| \|(X_{\hat{B}}^k)^{-1}(x_{\hat{B}}^{k+1} - x_{\hat{B}}^k)\| \\ &\leq \beta^2 - \|(X_{\hat{N}}^k)^{-1}(x_{\hat{N}}^{k+1} - x_{\hat{N}}^k)\|^2 + \|(X_{\hat{B}}^k)^{-1}(\hat{x}_{\hat{B}}^{k+1} - x_{\hat{B}}^{k+1})\|^2 \\ &\quad + 2\beta \|(X_{\hat{B}}^k)^{-1}(\hat{x}_{\hat{B}}^{k+1} - x_{\hat{B}}^{k+1})\|. \end{aligned}$$

By (46) and the fact that $\lim_{k \in \mathcal{K}} \Delta_{\hat{N}}^k = 0$, we have

$$(49) \quad \begin{aligned} \lim_{k \in \mathcal{K}_0} \|(X_{\hat{B}}^k)^{-1}(\hat{x}_{\hat{B}}^{k+1} - x_{\hat{B}}^{k+1})\| &\leq \lim_{k \in \mathcal{K}_0} \|(X_{\hat{B}}^k)^{-1}\| \|\hat{x}_{\hat{B}}^{k+1} - x_{\hat{B}}^{k+1}\| \\ &= \lim_{k \in \mathcal{K}_0} \mathcal{O}(\|(X_{\hat{B}}^k)^{-1}\| \|x_{\hat{N}}^k\|) = 0, \end{aligned}$$

where the last equality follows from (42) and (43). Combining (41), (48), and (49), we obtain (47).

For $\lambda \geq 0$, consider the point $x^{k+1}(\lambda) \equiv x^k + \lambda(\hat{x}^{k+1} - x^k)$. It follows from (47) that there exists $\epsilon > 0$ such that $x^{k+1}(\lambda) \in \mathcal{E}(x^k)$ for every $\lambda \in [0, 1 + \epsilon]$ and $k \in \mathcal{K}_0$ sufficiently large. We now estimate the difference $f(x^{k+1}) - f(x^{k+1}(\lambda))$. It is easy to see that

$$(50) \quad \begin{aligned} f(x^{k+1}) - f(x^{k+1}(\lambda)) &= f(x^{k+1}) - f(x^k) + f(x^k) - f(x^{k+1}(\lambda)) \\ &= \nabla f(x^k)^T (x^{k+1} - x^{k+1}(\lambda)) + \frac{1}{2} \|x^{k+1} - x^k\|_Q^2 \\ &\quad - \frac{1}{2} \|x^{k+1}(\lambda) - x^k\|_Q^2. \end{aligned}$$

Using (46) and the definition of $x^{k+1}(\lambda)$, we obtain

$$(51) \quad \begin{aligned} &\nabla f(x^k)^T (x^{k+1} - x^{k+1}(\lambda)) \\ &= (1 - \lambda) \nabla f(x^k)^T (x^{k+1} - x^k) + \lambda \nabla f(x^k)^T (x^{k+1} - \hat{x}^{k+1}) \\ &\geq (\lambda - 1) \nabla f(x^k)^T (x^k - x^{k+1}) - \mathcal{O}(\|x_{\hat{N}}^k\|) \\ &= (\lambda - 1) [\|x^k - x^{k+1}\|_Q^2 + \beta \|X^k s^k\|] - \mathcal{O}(\|x_{\hat{N}}^k\|), \end{aligned}$$

where the last equality follows from (12). Moreover,

$$(52) \quad \begin{aligned} \|x^{k+1}(\lambda) - x^k\|_Q^2 &= \lambda^2 \|\hat{x}^{k+1} - x^k\|_Q^2 \\ &\leq \lambda^2 \{ \|\hat{x}^{k+1} - x^{k+1}\|_Q^2 + \|x^{k+1} - x^k\|_Q^2 \\ &\quad + 2 \|\hat{x}^{k+1} - x^{k+1}\|_Q \|x^{k+1} - x^k\|_Q \} \\ &\leq \lambda^2 \|x^{k+1} - x^k\|_Q^2 + \mathcal{O}(\|x_{\hat{N}}^k\|) \quad \forall \lambda \in [0, 1 + \epsilon]. \end{aligned}$$

Combining (50), (51), and (52) and rearranging, we obtain that for every $\lambda \in (1, 1 + \epsilon]$,

$$(53) \quad \frac{f(x^{k+1}) - f(x^{k+1}(\lambda))}{(\lambda - 1) \|X^k s^k\|} \geq \beta - \frac{1}{2} (\lambda - 1) \frac{\|x^{k+1} - x^k\|_Q^2}{\|X^k s^k\|} - \mathcal{O} \left(\frac{\|x_{\hat{N}}^k\|}{(\lambda - 1) \|X^k s^k\|} \right).$$

Now, by (5), (40), and Lemma 4.2, it follows that

$$(54) \quad f(x^k) - f(x^*) \leq (1 - \beta)^{-1} [f(x^{k+1}) - f(x^*)] = \mathcal{O}(\|X^k s^k\|) \quad \forall k \in \mathcal{K} \supseteq \mathcal{K}_0.$$

This relation, relation (44) and the identity $f(x^k) - f(x^{k+1}) = \beta \|X^k s^k\| + \|x^{k+1} - x^k\|_Q^2/2$ imply that

$$(55) \quad \lim_{k \in \mathcal{K}_0} \frac{\|x_{\hat{N}}^k\|}{\|X^k s^k\|} = 0, \quad \frac{\|x^{k+1} - x^k\|_Q^2}{\|X^k s^k\|} \leq L_0 \quad \forall k \in \mathcal{K}_0,$$

where L_0 is some positive constant. Let $\bar{\lambda} \equiv 1 + \min\{\epsilon, \beta/L_0\}$. Then, it follows from (53) with $\lambda = \bar{\lambda}$ that

$$\frac{f(x^{k+1}) - f(x^{k+1}(\bar{\lambda}))}{(\bar{\lambda} - 1) \|X^k s^k\|} \geq \frac{\beta}{2} - \mathcal{O} \left(\frac{\|x_{\hat{N}}^k\|}{\|X^k s^k\|} \right) > 0 \quad \forall k \in \mathcal{K}_0 \text{ sufficiently large.}$$

Hence, $f(x^{k+1}(\bar{\lambda})) < f(x^{k+1})$ for every $k \in \mathcal{K}_0$ sufficiently large. This conclusion together with the fact that $x^{k+1}(\bar{\lambda}) \in \mathcal{E}(x^k)$ contradicts the definition of x^{k+1} . $\square$

The following set is frequently used in our presentation. Let

$$(56) \quad \Gamma \equiv \{N \mid N \subseteq N_* \subseteq \{1, \dots, n\}, \liminf_{k \rightarrow \infty} \Delta_N^k = 0\}.$$

We now give an outline for the remaining part of this section. The main result of this section is Theorem 4.15 which establishes global convergence of Algorithm AS. Its statement is very similar to the one of Theorem 3.6 except that it guarantees the existence of a unique partition $(\bar{B}, \bar{N})$ satisfying Local Assumption 3.2. Let $\bar{N}$ denote a minimal element of the set Γ with respect to inclusion and let $\bar{B} \equiv \{1, \dots, n\} \setminus \bar{N}$. The main goal from now on is to show that $(\bar{B}, \bar{N})$ satisfies all the conditions of Local Assumption 3.2. In Theorem 4.15 we show that there can be only one partition satisfying Local Assumption 3.2 from which it also follows that the set Γ has only one minimal element, in other words, Γ has a smallest element with respect to inclusion. Theorem 4.13 shows that $(\bar{B}, \bar{N})$ satisfies condition (A) of Local Assumption 3.2. In fact, once this condition is verified, it is not difficult to obtain the other ones. We separate the proof of Theorem 4.13 into several lemmas. The major part of the proof is given in Lemma 4.10, Lemma 4.11, and Lemma 4.12. All the results between (and including) Lemma 4.5 and 4.9 are mainly technical ones.

LEMMA 4.5. *Assume that an infinite index set $\mathcal{K}$ and an index set $N \subseteq \{1, \dots, n\}$ are such that $N \subseteq N_*$ and $\lim_{k \in \mathcal{K}} \Delta_N^k = 0$. Then, $\lim_{k \in \mathcal{K}} s_B^k = 0$, where $B \equiv \{1, \dots, n\} \setminus N$.*

Proof. By Theorem 4.4(a), it follows that there exists $L_0 > 0$ such that $f(x^k) - f(x^*) \leq L_0 \|x_N^k\|$ for every integer $k \geq 0$. Using this fact and inequality (12), we obtain

$$\begin{aligned} \|s_B^k\| &\leq \|(x_B^k)^{-1}\| \|x_B^k s_B^k\| \leq \|(x_B^k)^{-1}\| \|x^k s^k\| \\ &\leq \|(x_B^k)^{-1}\| \frac{f(x^k) - f(x^{k+1})}{\beta} \leq \|(x_B^k)^{-1}\| \frac{f(x^k) - f(x^*)}{\beta} \\ &\leq \frac{L_0}{\beta} \|(x_B^k)^{-1}\| \|x_N^k\| = \mathcal{O}(\Delta_N^k). \end{aligned}$$

Since by assumption $\lim_{k \in \mathcal{K}} \Delta_N^k = 0$, the result immediately follows. $\square$

LEMMA 4.6. *Suppose that $N \in \Gamma$ and $B \equiv \{1, \dots, n\} \setminus N$. Then the following statements hold:*

- (a) $g_B^* \in \text{Range}(A_B^T)$;
- (b) $x^* \in \mathcal{P}_N^+$ and $x \mapsto (g^*)^T x$ is constant over the face $\mathcal{P}_N^+$;
- (c) $\mathcal{P}_N^{++} \neq \emptyset$.

Proof. Since $N \in \Gamma$, we have $\liminf_{k \rightarrow \infty} \Delta_N^k = 0$. By Lemma 4.5, the sequence $\{(y^k, s^k)\}$ has an accumulation point $(\bar{y}, \bar{s})$ such that $\bar{s}_B = 0$. Since $A^T \bar{y} + \bar{s} = g^*$ and $\bar{s}_B = 0$, (a) follows. Statement (b) follows from (a) and the fact that $N \subseteq N_*$. We next show (c). Let $\mathcal{K}$ be an infinite index set such that $\lim_{k \in \mathcal{K}} \Delta_N^k = 0$. Consider the feasible system (e.g., $x = x^*$ is a feasible solution)

$$(57) \quad Ax = b, \quad x_N = 0.$$

Using Lemma 4.1 with $z = x^k$, there exist a constant $L_0 > 0$ and a sequence $\{w^k\}$ of solutions of (57) such that

$$\|x^k - w^k\| \leq L_0 \|x_N^k\| \quad \forall k \geq 0.$$

Hence, we obtain

$$\|(X_B^k)^{-1}(w_B^k - x_B^k)\| \leq \|(x_B^k)^{-1}\| \|w_B^k - x_B^k\| \leq L_0 \|(x_B^k)^{-1}\| \|x_N^k\| = L_0 \Delta_B^k,$$

from which we conclude that $\lim_{k \in \mathcal{K}} \|(X_B^k)^{-1}(w_B^k - x_B^k)\| = 0$. Clearly, this implies that $w_B^k > 0$ for sufficiently large $k \in \mathcal{K}$, and hence that $w^k \in \mathcal{P}_N^{++}$, for all $k \in \mathcal{K}$ sufficiently large. $\square$

LEMMA 4.7. *Assume that $\bar{N}$ is a minimal element of Γ with respect to inclusion and let $\bar{B} = \{1, \dots, n\} \setminus \bar{N}$. Then there exist scalars $\delta_0 > 0$ and $\eta_0 > 0$ such that the following implication holds for every k sufficiently large:*

$$(58) \quad \Delta_{\bar{N}}^k \leq \delta_0 \implies \frac{\|X_{\bar{N}}^k s_{\bar{N}}^k\|}{\|x_{\bar{N}}^k\|} \geq \eta_0.$$

Proof. Assume for contradiction that the conclusion of the lemma does not hold. Then there exists an infinite index set $\mathcal{K}$ such that

$$(59) \quad \lim_{k \in \mathcal{K}} \Delta_{\bar{N}}^k = 0, \quad \lim_{k \in \mathcal{K}} \frac{\|X_{\bar{N}}^k s_{\bar{N}}^k\|}{\|x_{\bar{N}}^k\|} = 0.$$

Since $\{s_{\bar{N}}^k\}$ and $\{x_{\bar{N}}^k/\|x_{\bar{N}}^k\|\}$ are bounded sequences, by passing to a subsequence if necessary we may assume that

$$(60) \quad \lim_{k \in \mathcal{K}} s_{\bar{N}}^k = s_{\bar{N}}^\infty, \text{ and } \lim_{k \in \mathcal{K}} \frac{x_{\bar{N}}^k}{\|x_{\bar{N}}^k\|} = d_{\bar{N}}^\infty$$

for some $|\bar{N}|$ -dimensional vectors $s_{\bar{N}}^\infty$ and $d_{\bar{N}}^\infty$. Clearly, $d_{\bar{N}}^\infty \geq 0$, $\|d_{\bar{N}}^\infty\| = 1$ and, by the second relation in (59), $S_{\bar{N}}^\infty d_{\bar{N}}^\infty = 0$. Let $N \equiv \{j \in \bar{N} \mid d_j^\infty = 0\}$ and $B \equiv \{1, \dots, n\} \setminus N$. Since $\|d_{\bar{N}}^\infty\| = 1$ and $s_{\bar{N}}^\infty \neq 0$, it follows that $N \subseteq \bar{N}$ properly, and hence that $B \supseteq \bar{B}$ properly. Using the first relation in (59), the second relation in (60), and the fact that $d_N^\infty = 0$ and $d_{N \setminus N}^\infty > 0$, we can easily see that $\lim_{k \in \mathcal{K}} \Delta_N^k = 0$. Hence, it follows that $\liminf_{k \rightarrow \infty} \Delta_N^k = 0$, and since $N \subseteq \bar{N} \subseteq N_*$, we conclude that $N \in \Gamma$. But this together with the fact that N is contained $\bar{N}$ properly contradict the assumption that $\bar{N}$ is a minimal element of Γ . $\square$

The next result plays an important role in several of the results that will be stated later. It is useful to recall the definition of the set $\mathcal{D}_B(x)$ given in (3).

LEMMA 4.8. *The following statements hold:*

- (a) $s_{B_*}^k = \mathcal{O}(\|X^k s^k\| \|x^{k+1} - x^k\|)$;
- (b) *there exists a sequence $\{(y_*^k, s_*^k)\} \subseteq \mathcal{D}_{B_*}(x^*)$ such that*

$$(61) \quad \|s_{N_*}^k - (s_*^k)_{N_*}\| = \mathcal{O}(\|x^{k+1} - x^*\|),$$

$$(62) \quad \|X^k(s^k - s_*^k)\| = \mathcal{O}(\|X^k s^k\|).$$

Proof. Statement (a) follows from the fact that $(x_i^k)^{-1}$ is bounded above for every $i \in B_*$ and from the relation

$$s_i^k = (x_i^k)^{-2}(x_i^k - x_i^{k+1}) \frac{\|X^k s^k\|}{\beta}.$$

We next show (b). Consider the feasible system

$$(63) \quad A^T y + s = g^*, \quad s_{B_*} = 0$$

(e.g., if (y^*, s^*) is an accumulation point of $\{(y^k, s^k)\}$ then $(y, s) = (y^*, s^*)$ is a feasible solution of this system). Using Lemma 4.1 with $z = (y^k, s^k)$ and the fact that $A^T y^k + s^k = g^{k+1}$, we conclude that for every $k \geq 0$, (63) has a solution (y_*^k, s_*^k) such that

$$\|y^k - y_*^k\| + \|s^k - s_*^k\| = \mathcal{O}(\|g^{k+1} - g^*\| + \|s_{B_*}^k\|).$$

Clearly, $\{(y_*^k, s_*^k)\} \subseteq \mathcal{D}_{B_*}(x^*)$, and (61) now follows from the above expression, statement (a) and the fact that $\|g^{k+1} - g^*\| = \|Q(x^{k+1} - x^*)\| = \mathcal{O}(\|x^{k+1} - x^*\|)$. We next show (62). First observe that (a) and the fact that $(s_*^k)_{B_*} = 0$ imply that

$$\|X_{B_*}^k (s^k - s_*^k)_{B_*}\| = \|X_{B_*}^k s_{B_*}^k\| = \mathcal{O}(\|x^{k+1} - x^k\| \|X^k s^k\|) \leq \mathcal{O}(\|X^k s^k\|).$$

By Lemma 4.2 and Lemma 2.4(c), it follows that

$$\|x^{k+1} - x^*\|^2 = \mathcal{O}(\|X^k s^k\|).$$

Using this relation, (5) and (61), we obtain

$$\begin{aligned} \|X_{N_*}^k (s^k - s_*^k)_{N_*}\| &\leq (1 - \beta)^{-1} \|x_{N_*}^{k+1}\| \|(s^k - s_*^k)_{N_*}\| \\ &\leq \mathcal{O}(\|x^{k+1} - x^*\|^2) = \mathcal{O}(\|X^k s^k\|). \quad \square \end{aligned}$$

LEMMA 4.9. *Let (B, N) be a partition such that $\mathcal{D}_B(x^*) \neq \emptyset$. Then there exists a constant $C_3 > 0$ such that $\|(P(x)Xg^*)_B\| \leq C_3 (\|x_N\| \Delta_N(x))$ for every $x \in \mathcal{P}_{++}$ with $\Delta_N(x)$ sufficiently small.*

Proof. Since $\mathcal{D}_B(x^*) \neq \emptyset$, let $(\bar{y}, \bar{s}) \in \mathcal{D}_B(x^*)$ be given. We have $\bar{s}_B = 0$ and $P(x)Xg^* = P(x)X\bar{s}$. From Lemma 3.1, it follows that

$$P(x)X\bar{s} = \begin{pmatrix} \tilde{P}_N(x_N)X_N\bar{s}_N \\ \tilde{P}_B(x_B)X_B\bar{s}_B \end{pmatrix} + R(x)X\bar{s},$$

where $\|R(x)\| \leq C_1 \Delta_N(x)$ for every $x \in \mathcal{P}_{++}$ with $\Delta_N(x)$ sufficiently small. Since $\bar{s}_B = 0$, we obtain

$$\|(P(x)Xg^*)_B\| = \|(P(x)X\bar{s})_B\| \leq \|R(x)\| \|X_N\bar{s}_N\| \leq C_1 \|\bar{s}_N\| \|x_N\| \Delta_N(x),$$

for every $x \in \mathcal{P}_{++}$ with $\Delta_N(x)$ sufficiently small. Letting $C_3 \equiv C_1 \|\bar{s}_N\|$, the result follows. $\square$

The main goal of the next three results is to prepare the reader for the proof that $\sum_{k=0}^{\infty} \Delta_N^k < \infty$. This fact will be more easily proved by working with a related sequence of scalars, namely the sequence $\{\Psi_B^k\}$ defined as

$$(64) \quad \Psi_B^k \equiv \frac{f(x^k) - f(x^*)}{\min(x_B^k)} \quad \forall k \geq 0.$$

We will show that $\sum_{k=0}^{\infty} \Psi_B^k < \infty$, which in turn allows us to show that $\sum_{k=0}^{\infty} \Delta_N^k < \infty$. For Δ_N^k sufficiently small, the next three lemmas deal with three cases (not necessarily disjoint) that arise by considering the size of the two quantities:

$$\frac{\|x_{N_* \setminus \bar{N}}^k\|^2}{\|x_{\bar{N}}^k\|}, \quad \frac{\min(x_{N_* \setminus \bar{N}}^k)}{\max(x_{N_* \setminus \bar{N}}^k)}.$$

Lemma 4.10 deals with the case where the first quantity is sufficiently small; Lemma 4.11 deals with the case where the second quantity is sufficiently small; finally, Lemma 4.12 deals with the case where the first and second quantities are bounded away from 0. In the proof of Theorem 4.13, we combine these three cases together to prove that $\sum_{k=0}^{\infty} \Delta_{\bar{N}}^k < \infty$.

LEMMA 4.10. *Assume that $\bar{N}$ is a minimal element of Γ with respect to inclusion and let $\bar{B} = \{1, \dots, n\} \setminus \bar{N}$. There exist constants $\delta_1 > 0$, $\tau_1 > 0$ and $\gamma_1 \in (0, 1)$ such that the following implication holds for every k sufficiently large:*

$$\|x_{N_* \setminus \bar{N}}^k\|^2 \leq \tau_1 \|x_{\bar{N}}^k\|, \quad \Delta_{\bar{N}}^k \leq \delta_1 \implies \Psi_{\bar{B}}^{k+1} \leq \gamma_1 \Psi_{\bar{B}}^k.$$

Proof. Assume for contradiction that for some infinite index set $\mathcal{K}$, there holds

$$(65) \quad \lim_{k \in \mathcal{K}} \frac{\|x_{N_* \setminus \bar{N}}^k\|^2}{\|x_{\bar{N}}^k\|} = 0,$$

$$(66) \quad \lim_{k \in \mathcal{K}} \Delta_{\bar{N}}^k = 0,$$

$$(67) \quad \limsup_{k \in \mathcal{K}} \frac{\Psi_{\bar{B}}^{k+1}}{\Psi_{\bar{B}}^k} \geq 1.$$

Observe that (66) and Lemma 4.7 imply that

$$(68) \quad \|x_{\bar{N}}^k\| = \mathcal{O}(\|X^k s^k\|) \quad \forall k \in \mathcal{K}.$$

Below we concentrate our effort in showing the following claim:

$$(69) \quad \lim_{k \in \mathcal{K}} (X_{\bar{B}}^k)^{-1} (x_{\bar{B}}^{k+1} - x_{\bar{B}}^k) = 0.$$

This relation clearly implies that

$$\lim_{k \in \mathcal{K}} \frac{\min(x_{\bar{B}}^{k+1})}{\min(x_{\bar{B}}^k)} = 1,$$

which together with Lemma 2.4(b) in turn imply that $\limsup_{k \in \mathcal{K}} \Psi_{\bar{B}}^{k+1} / \Psi_{\bar{B}}^k < 1$, a fact that contradicts (67) and hence establishes the validity of the lemma.

To prove the claim, first observe that

$$(70) \quad (X^k)^{-1} (x^{k+1} - x^k) = \beta(p_1^k + p_2^k),$$

where

$$(71) \quad p_1^k \equiv P(x^k) \frac{X^k g^*}{\|X^k s^k\|}, \quad p_2^k \equiv P(x^k) \frac{X^k (g^{k+1} - g^*)}{\|X^k s^k\|}.$$

The claim follows once we show that $\lim_{k \in \mathcal{K}} (p_1^k)_{\bar{B}} = 0$ and $\lim_{k \in \mathcal{K}} p_2^k = 0$. We first show that $\lim_{k \in \mathcal{K}} (p_1^k)_{\bar{B}} = 0$. Since $\bar{N} \in \Gamma$, it follows from Lemma 4.6(a) that $g_{\bar{B}}^* \in \text{Range}(A_{\bar{B}}^T)$, or equivalently, that $\mathcal{D}_{\bar{B}}(x^*) \neq \emptyset$. Hence, by (66), (68), and Lemma 4.9, we obtain

$$(72) \quad (p_1^k)_{\bar{B}} = \mathcal{O} \left(\frac{\|x_{\bar{N}}^k\| \Delta_{\bar{N}}^k}{\|X^k s^k\|} \right) \leq \mathcal{O}(\Delta_{\bar{N}}^k) \quad \forall k \in \mathcal{K},$$

from which we conclude that $\lim_{k \in \mathcal{K}} (p_1^k)_{\bar{B}} = 0$. We next show that $\lim_{k \in \mathcal{K}} p_2^k = 0$. Indeed, recalling the definition of (y_*^k, s_*^k) in Lemma 4.8(b) and using the fact that $(y^k, s^k) \in \mathcal{D}(x^{k+1})$ and $(y_*^k, s_*^k) \in \mathcal{D}(x^*)$, we obtain

$$p_2^k = P(x^k) \frac{X^k(s^k - s_*^k)}{\|X^k s^k\|}$$

and hence,

$$(73) \quad \|p_2^k\| \leq \frac{\|X^k(s^k - s_*^k)\|}{\|X^k s^k\|}.$$

Now, Lemma 4.8(a) and the fact that $(s^k)_{B_*} = 0$ imply

$$(74) \quad \frac{\|X_{B_*}^k(s_{B_*}^k - (s_*^k)_{B_*})\|}{\|X^k s^k\|} \leq \|x_{B_*}^k\| \frac{\|s_{B_*}^k\|}{\|X^k s^k\|} = \mathcal{O}(\|x^{k+1} - x^k\|) \quad \forall k \in \mathcal{K},$$

and, from Lemma 4.8(b), Theorem 4.4(b), and (68), we obtain

$$(75) \quad \begin{aligned} \frac{\|X_{N_*}^k(s_{N_*}^k - (s_*^k)_{N_*})\|}{\|X^k s^k\|} &\leq \frac{(\|x_{\bar{N}}^k\| + \|x_{N_* \setminus \bar{N}}^k\|) \|s_{N_*}^k - (s_*^k)_{N_*}\|}{\|X^k s^k\|} \\ &\leq \mathcal{O} \left(\frac{(\|x_{\bar{N}}^k\| + \|x_{N_* \setminus \bar{N}}^k\|)}{\|x_{\bar{N}}^k\|^{1/2}} \frac{\|x^{k+1} - x^*\|}{\|x_{\bar{N}}^k\|^{1/2}} \right) \\ &\leq \mathcal{O} \left(\frac{\|x_{\bar{N}}^k\| + \|x_{N_* \setminus \bar{N}}^k\|}{\|x_{\bar{N}}^k\|^{1/2}} \right) \quad \forall k \in \mathcal{K}, \end{aligned}$$

which converges to 0 in view of (65). It follows from (73), (74), and (75) that $\lim_{k \in \mathcal{K}} p_2^k = 0$. We have thus shown that $\lim_{k \in \mathcal{K}} (X_B^k)^{-1}(x_B^{k+1} - x_B^k) = 0$. $\square$

LEMMA 4.11. *Assume that $\bar{N}$ is a minimal element of Γ with respect to inclusion and let $\bar{B} = \{1, \dots, n\} \setminus \bar{N}$. There exist constants $\delta_2 > 0$, $M_0 > 0$ and $\gamma_2 \in (0, 1)$ such that the following implication holds for every k sufficiently large:*

$$\frac{\max(x_{N_* \setminus \bar{N}}^k)}{\min(x_{N_* \setminus \bar{N}}^k)} \geq M_0, \quad \Delta_{\bar{N}}^k \leq \delta_2 \implies \Psi_{\bar{B}}^{k+1} \leq \gamma_2 \Psi_{\bar{B}}^k.$$

Proof. Assume for contradiction that for some infinite index set $\mathcal{K}$, there holds

$$(76) \quad \lim_{k \in \mathcal{K}} \frac{\max(x_{N_* \setminus \bar{N}}^k)}{\min(x_{N_* \setminus \bar{N}}^k)} = \infty,$$

$$(77) \quad \lim_{k \in \mathcal{K}} \Delta_{\bar{N}}^k = 0,$$

$$(78) \quad \limsup_{k \in \mathcal{K}} \frac{\Psi_{\bar{B}}^{k+1}}{\Psi_{\bar{B}}^k} \geq 1.$$

It is easy to see that there exist some infinite subset $\mathcal{K}_0$ of $\mathcal{K}$ and some set $N \subseteq \{1, \dots, n\}$ such that

$$(79) \quad \limsup_{k \in \mathcal{K}_0} \frac{x_i^k}{\min(x_{N_* \setminus \bar{N}}^k)} < \infty \quad \forall i \in N,$$

$$(80) \quad \lim_{k \in \mathcal{K}_0} \frac{x_i^k}{\min(x_{N_* \setminus \bar{N}}^k)} = \infty \quad \forall i \notin N.$$

Using (77) and the fact that $\lim_{k \rightarrow \infty} x_{B_*}^k > 0$, it is easy to see that $\bar{N} \subseteq N \subseteq N_*$. Moreover, the definition of N , (76), and (77) imply that these two inclusions are proper. We claim that

$$(81) \quad \lim_{k \in \mathcal{K}_0} (X_{N \setminus \bar{N}}^k)^{-1} (x_{N \setminus \bar{N}}^{k+1} - x_{N \setminus \bar{N}}^k) = 0.$$

As we will see, showing this relation is the major effort of the proof. As in (70), we know that $(X^k)^{-1}(x^{k+1} - x^k) = \beta(p_1^k + p_2^k)$, where p_1^k and p_2^k are defined in (71). The claim follows if we show that $\lim_{k \in \mathcal{K}} (p_1^k)_{\bar{B}} = 0$ and $\lim_{k \in \mathcal{K}_0} (p_2^k)_N = 0$. The proof that $\lim_{k \in \mathcal{K}} (p_1^k)_{\bar{B}} = 0$ is exactly the same as in (72) of Lemma 4.10. We next show that $\lim_{k \in \mathcal{K}_0} (p_2^k)_N = 0$. First observe that (79) and (80) imply that $\lim_{k \in \mathcal{K}_0} \Delta_N^k = 0$. Let $B \equiv \{1, \dots, n\} \setminus N$. Using Lemma 3.1, we obtain

$$\begin{aligned} p_2^k &= \frac{1}{\|X^k s^k\|} P(x^k) X^k (s^k - s_*^k) \\ &= \frac{1}{\|X^k s^k\|} \begin{pmatrix} \tilde{P}_N(x_N^k) X_N^k (s^k - s_*^k)_N \\ \tilde{P}_B(x_B^k) X_B^k (s^k - s_*^k)_B \end{pmatrix} + R(x^k) \frac{X^k (s^k - s_*^k)}{\|X^k s^k\|}, \end{aligned}$$

where $\|R(x^k)\| = \mathcal{O}(\Delta_N^k)$. Using this relation, (61), (62), and (5), we obtain

$$\begin{aligned} \|(p_2^k)_N\| &\leq \frac{\|\tilde{P}_N(x_N^k) X_N^k (s^k - s_*^k)_N\|}{\|X^k s^k\|} + \|R(x^k)\| \frac{\|X^k (s^k - s_*^k)\|}{\|X^k s^k\|} \\ &\leq \frac{\|x_N^k\| \|(s^k - s_*^k)_N\|}{\|X^k s^k\|} + \mathcal{O}(\Delta_N^k) \\ &\leq \mathcal{O} \left(\frac{\|x_N^k\|}{\|x^{k+1} - x^*\|} \frac{\|x^{k+1} - x^*\|^2}{\|X^k s^k\|} \right) + \mathcal{O}(\Delta_N^k) \\ &\leq \mathcal{O} \left(\frac{\|x_N^k\|}{\|x_{N_* \setminus N}^{k+1}\|} \right) + \mathcal{O}(\Delta_N^k) \\ &\leq \mathcal{O} \left(\frac{\|x_N^k\|}{\|x_{N_* \setminus N}^k\|} \right) + \mathcal{O}(\Delta_N^k) \\ &\leq \mathcal{O}(\Delta_N^k), \end{aligned}$$

where in the third inequality we used the fact that $\|x^{k+1} - x^*\|^2 = \mathcal{O}(\|X^k s^k\|)$, due to Lemma 2.4(c) and Lemma 4.2. Since $\{\Delta_N^k\}_{k \in \mathcal{K}_0}$ converges to 0, the last expression shows that $\lim_{k \in \mathcal{K}_0} (p_2^k)_N = 0$. Using (5), (79), and (80), it is easy to see that for every $k \in \mathcal{K}_0$ sufficiently large,

$$(82) \quad \min(x_{\bar{B}}^k) = \min(x_{N \setminus \bar{N}}^k), \quad \min(x_{\bar{B}}^{k+1}) = \min(x_{N \setminus \bar{N}}^{k+1}).$$

It then follows from (81) and (82) that

$$\lim_{k \in \mathcal{K}_0} \left(\frac{\min(x_{\bar{B}}^{k+1})}{\min(x_{\bar{B}}^k)} \right) = \lim_{k \in \mathcal{K}_0} \left(\frac{\min(x_{N \setminus \bar{N}}^{k+1})}{\min(x_{N \setminus \bar{N}}^k)} \right) = 1.$$

This relation and Lemma 2.4(b) then imply that $\limsup_{k \in \mathcal{K}_0} \Psi_{\bar{B}}^{k+1} / \Psi_{\bar{B}}^k < 1$, a fact that contradicts (78). $\square$

LEMMA 4.12. Let $\bar{B} \equiv \{1, \dots, n\} \setminus \bar{N}$, where $\bar{N}$ is a minimal element of Γ . Let δ_0 , τ_1 and M_0 be the constants introduced in Lemmas 4.7, 4.10, and 4.11. Then there exist constants $\alpha_1 > 0$, $\alpha_2 > 0$, and $M_1 > 0$ such that the following two implications hold for every k sufficiently large:

$$(83) \quad \Delta_{\bar{N}}^k \leq \delta_0 \implies \alpha_1 \Psi_{\bar{B}}^k \leq \Delta_{\bar{N}}^k \leq \alpha_2 \Psi_{\bar{B}}^k,$$

$$(84) \quad \left. \begin{array}{l} \Delta_{\bar{N}}^k \leq \delta_0, \\ \|x_{N_* \setminus \bar{N}}^k\|^2 \geq \tau_1 \|x_{\bar{N}}^k\|, \\ \max(x_{N_* \setminus \bar{N}}^k) \leq M_0 \min(x_{N_* \setminus \bar{N}}^k) \end{array} \right\} \implies \Delta_{\bar{N}}^k \leq M_1 (f(x^k) - f(x^*))^{1/2}.$$

Proof. We first show implication (83). Assume that k_0 is such that implication (58) holds for every $k \geq k_0$. Using (58), (12), and Theorem 4.4(a), we conclude that if $k \geq k_0$ and $\Delta_{\bar{N}}^k \leq \delta_0$ then

$$\begin{aligned} \|x_{\bar{N}}^k\| &\leq \frac{\|X_{\bar{N}}^k s_{\bar{N}}^k\|}{\eta_0} \leq \frac{\|X^k s^k\|}{\eta_0} \leq \frac{f(x^k) - f(x^{k+1})}{\beta \eta_0} \\ &\leq \frac{f(x^k) - f(x^*)}{\beta \eta_0} = \mathcal{O}(\|x_{\bar{N}}^k\|), \end{aligned}$$

or equivalently, that $\|x_{\bar{N}}^k\| \sim f(x^k) - f(x^*)$, for every k such that $\Delta_{\bar{N}}^k \leq \delta_0$. Using this fact, it is now straightforward to verify that there exist $\alpha_1 > 0$ and $\alpha_2 > 0$ such that (83) holds for every $k \geq k_0$. We now show that (84) holds. Assume that $k \geq k_0$ is such that $\Delta_{\bar{N}}^k \leq \delta_0$, $\|x_{N_* \setminus \bar{N}}^k\|^2 \geq \tau_1 \|x_{\bar{N}}^k\|$ and $\max(x_{N_* \setminus \bar{N}}^k) \leq M_0 \min(x_{N_* \setminus \bar{N}}^k)$. Then,

$$\begin{aligned} \|(x_{\bar{B}}^k)^{-1}\| \|x_{\bar{N}}^k\| &\leq \frac{\max((x_{\bar{B}}^k)^{-1}) \|x_{\bar{N}}^k\|}{\sqrt{n}} = \frac{\|x_{\bar{N}}^k\|}{\sqrt{n} \min(x_{\bar{B}}^k)} \\ &\leq \frac{\|x_{\bar{N}}^k\|}{\sqrt{n} \min(x_{N_* \setminus \bar{N}}^k)} \leq \frac{M_0 \|x_{\bar{N}}^k\|}{\sqrt{n} \max(x_{N_* \setminus \bar{N}}^k)} \\ &\leq \frac{M_0 \|x_{\bar{N}}^k\|}{\|x_{N_* \setminus \bar{N}}^k\|} \leq \frac{M_0 \|x_{\bar{N}}^k\|}{\sqrt{\tau_1} \|x_{\bar{N}}^k\|} \\ &= \frac{M_0}{\sqrt{\tau_1}} \|x_{\bar{N}}^k\|^{1/2} = \mathcal{O}(f(x^k) - f(x^*))^{1/2}. \end{aligned}$$

This shows that there exists $M_1 > 0$ such that (84) holds for every $k \geq k_0$. $\square$

The following result shows that the partition $(\bar{B}, \bar{N})$, where $\bar{N}$ is a minimal element of Γ , satisfies condition (A) of Assumption 3.2.

THEOREM 4.13. Let $\bar{B} \equiv \{1, \dots, n\} \setminus \bar{N}$, where $\bar{N}$ is a minimal element of Γ . Then, $\sum_{k=0}^{\infty} \Delta_{\bar{N}}^k < \infty$.

Proof. First of all, it is useful to recall the content of Lemmas 4.10, 4.11, and 4.12 here. Namely, there exist positive constants δ_0 , δ_1 , δ_2 , M_0 , M_1 , τ_1 , α_1 , α_2 , $\gamma_1 < 1$, and $\gamma_2 < 1$ such that the following implications hold:

$$(85) \quad \|x_{N_* \setminus \bar{N}}^k\|^2 \leq \tau_1 \|x_{\bar{N}}^k\|, \quad \Delta_{\bar{N}}^k \leq \delta_1 \implies \Psi_{\bar{B}}^{k+1} \leq \gamma_1 \Psi_{\bar{B}}^k;$$

$$(86) \quad \frac{\max(x_{N_* \setminus \bar{N}}^k)}{\min(x_{N_* \setminus \bar{N}}^k)} \geq M_0, \quad \Delta_{\bar{N}}^k \leq \delta_2 \implies \Psi_{\bar{B}}^{k+1} \leq \gamma_2 \Psi_{\bar{B}}^k;$$

$$(87) \quad \Delta_N^k \leq \delta_0 \implies \alpha_1 \Psi_B^k \leq \Delta_N^k \leq \alpha_2 \Psi_B^k;$$

$$(88) \quad \left. \begin{array}{l} \Delta_N^k \leq \delta_0, \\ \|x_{N_* \setminus N}^k\|^2 \geq \tau_1 \|x_N^k\|, \\ \max(x_{N_* \setminus N}^k) \leq M_0 \min(x_{N_* \setminus N}^k) \end{array} \right\} \implies \Delta_N^k \leq M_1(f(x^k) - f(x^*))^{1/2}.$$

Observe that implications (85), (86), and (88) cover all possible situations for Δ_N^k sufficiently small. Define

$$(89) \quad \epsilon_1 \equiv \frac{1-\beta}{1+\beta} \min\{\delta_0, \delta_1, \delta_2\}.$$

First, we claim that there exists an integer $k_0 \geq 0$ such that

$$(90) \quad \Delta_N^k \leq \epsilon_1 \quad \forall k \geq k_0.$$

Indeed, let

$$(91) \quad \epsilon_2 \equiv \frac{(1-\beta)\epsilon_1}{(1+\beta)\alpha_2}, \quad \epsilon_3 \equiv \left(\frac{\alpha_1(1-\beta)\epsilon_2}{M_1} \right)^2.$$

Due to Lemmas 4.7, 4.10, 4.11, and 4.12 and the fact that $\liminf_{k \rightarrow \infty} \Delta_N^k = 0$ and $\lim_{k \rightarrow \infty} f(x^k) = f(x^*)$, it is easy to show the existence of an integer $k_0 \geq 0$ such that the implications (85), (86), (87), and (88) hold for every $k \geq k_0$ and

$$(92) \quad \Delta_N^{k_0} \leq \epsilon_1, \quad \Psi_B^{k_0} \leq \epsilon_2, \quad f(x^{k_0}) - f(x^*) \leq \epsilon_3,$$

where we use (88) to obtain the second relation. We will prove by induction on k that

$$(93) \quad \Delta_N^k \leq \epsilon_1, \quad \Psi_B^k \leq \epsilon_2,$$

for all $k \geq k_0$, thereby proving our claim (90). In view of the first two inequalities in (92), k_0 satisfies (93). Assume that the index k satisfies (93). Since $\Delta_N^k \leq \epsilon_1 \leq \min\{\delta_0, \delta_1, \delta_2\}$ and the implications (85), (86), (87), and (88) hold, we conclude that

$$(94) \quad \alpha_1 \Psi_B^k \leq \Delta_N^k \leq \alpha_2 \Psi_B^k,$$

and that either

$$(95) \quad \Delta_N^k \leq M_1(f(x^k) - f(x^*))^{1/2}$$

holds, or the inequality

$$(96) \quad \Psi_B^{k+1} \leq \bar{\gamma} \Psi_B^k$$

holds, where $\bar{\gamma} \equiv \max\{\gamma_1, \gamma_2\} < 1$. Using (5), (94), the second inequality in (93), and (91), we obtain

$$\Delta_N^{k+1} \leq \frac{1+\beta}{1-\beta} \Delta_N^k \leq \frac{1+\beta}{1-\beta} \alpha_2 \Psi_B^k \leq \frac{1+\beta}{1-\beta} \alpha_2 \epsilon_2 = \epsilon_1.$$

If inequality (95) holds, then, using (5), (94), the last inequality in (92), (91), and the fact that $\{f(x^k) - f(x^*)\}$ decreases monotonically, we obtain

$$\begin{aligned} \Psi_B^{k+1} &\leq \frac{1}{1-\beta} \Psi_B^k \leq \frac{1}{\alpha_1(1-\beta)} \Delta_N^k \\ &\leq \frac{M_1}{\alpha_1(1-\beta)} (f(x^k) - f(x^*))^{1/2} \leq \frac{M_1}{\alpha_1(1-\beta)} \epsilon_3^{1/2} = \epsilon_2. \end{aligned}$$

If instead (96) holds, then, by the last inequality of (93) and the fact that $\bar{\gamma} < 1$, we obtain $\Psi_{\bar{B}}^{k+1} \leq \bar{\gamma} \Psi_{\bar{B}}^k \leq \Psi_{\bar{B}}^k \leq \epsilon_2$. We have thus shown that $\Delta_{\bar{N}}^{k+1} \leq \epsilon_1$ and $\Psi_{\bar{B}}^{k+1} \leq \epsilon_2$, and hence that (93) holds for every $k \geq k_0$.

We now show that $\sum_{k=0}^{\infty} \Delta_{\bar{N}}^k < \infty$. Indeed, since, for every $k \geq k_0$, $\Delta_{\bar{N}}^k \leq \min\{\delta_0, \delta_1, \delta_2\}$ and the implications (85), (86), (87), and (88) hold, it follows that (94) holds and that either (95) or (96) holds, for every $k \geq k_0$. Since (94) holds for every $k \geq k_0$, it is sufficient to show that $\sum_{k=0}^{\infty} \Psi_{\bar{B}}^k < \infty$. Let $\mathcal{K}$ denote those indices $k \geq k_0$ for which (95) holds. If $\mathcal{K}$ is finite, then we have $\sum_{k=0}^{\infty} \Psi_{\bar{B}}^k < \infty$ since (96) holds for every k sufficiently large. So assume that $\mathcal{K}$ is infinite and, without loss of generality, that $k_0 \in \mathcal{K}$. Note that if $k' < k''$ are two consecutive indices in $\mathcal{K}$ then

$$\sum_{k=k'+1}^{k''} \Psi_{\bar{B}}^k \leq \frac{1}{1-\bar{\gamma}} \Psi_{\bar{B}}^{k'+1},$$

since (96) holds for every $k = k' + 1, \dots, k'' - 1$. Using this observation, (5), (94), the fact that (95) holds for every $k \in \mathcal{K}$, and Lemma 2.4(b), we obtain

$$\begin{aligned} \sum_{k=k_0+1}^{\infty} \Psi_{\bar{B}}^k &\leq \frac{1}{1-\bar{\gamma}} \sum_{k \in \mathcal{K}} \Psi_{\bar{B}}^{k+1} \leq \frac{1}{(1-\beta)(1-\bar{\gamma})} \sum_{k \in \mathcal{K}} \Psi_{\bar{B}}^k \\ &\leq \frac{1}{\alpha_1(1-\beta)(1-\bar{\gamma})} \sum_{k \in \mathcal{K}} \Delta_{\bar{N}}^k \\ &\leq \frac{M_1}{\alpha_1(1-\beta)(1-\bar{\gamma})} \sum_{k \in \mathcal{K}} (f(x^k) - f(x^*))^{1/2} \\ &\leq \frac{M_1}{\alpha_1(1-\beta)(1-\bar{\gamma})} \sum_{k=0}^{\infty} (f(x^k) - f(x^*))^{1/2} < \infty. \end{aligned}$$

We have thus proved that $\sum_{k=0}^{\infty} \Psi_{\bar{B}}^k < \infty$, and hence that $\sum_{k=0}^{\infty} \Delta_{\bar{N}}^k < \infty$. $\square$

The following lemma is the last step toward the main result of this paper.

LEMMA 4.14. *Let $\bar{B} \equiv \{1, \dots, n\} \setminus \bar{N}$ where $\bar{N}$ is a minimal element of Γ . Then the following statements hold:*

(a) *For every $i \in \bar{N}$, there holds*

$$x_i^k \sim \|x_{\bar{N}}^k\| \sim \|X^k s^k\| \sim \|X_{\bar{N}}^k s_{\bar{N}}^k\| \sim f(x^k) - f(x^*) \sim f(x^k) - f(x^{k+1});$$

(b) $\lim_{k \rightarrow \infty} \|X_{\bar{N}}^k (s^k - \tilde{s}^k)_{\bar{N}}\| / \|x_{\bar{N}}^k\| = 0$;

(c) $\liminf_{k \rightarrow \infty} (g^*)^T (x^k - x^{k+1}) / \|x_{\bar{N}}^k\| > 0$.

Proof. Due to relation (12), Lemma 4.7 and Theorem 4.4(a), we conclude that for every $i \in \bar{N}$ and k sufficiently large,

$$\begin{aligned} x_i^k &\leq \|x_{\bar{N}}^k\| \leq \frac{\|X_{\bar{N}}^k s_{\bar{N}}^k\|}{\eta_0} \leq \frac{\|X^k s^k\|}{\eta_0} \leq \frac{f(x^k) - f(x^{k+1})}{\beta \eta_0} \\ &\leq \frac{f(x^k) - f(x^*)}{\beta \eta_0} = \mathcal{O}(x_i^k), \end{aligned}$$

which clearly implies statement (a). To prove (b), note that by (7) and (18), we have

$$X^k (s^k - \tilde{s}^k) = P(x^k) X^k (g^{k+1} - g^*) = P(x^k) X^k (s^k - s_*^k),$$

where $\{s_*^k\}$ is the sequence introduced in Lemma 4.8. Using Lemma 3.1, we obtain

$$X^k(s^k - \tilde{s}^k) = \begin{pmatrix} \tilde{P}_{\bar{B}}(x_{\bar{B}}^k)X_{\bar{B}}^k(s^k - s_*^k)_{\bar{B}} \\ \tilde{P}_{\bar{N}}(x_{\bar{N}}^k)X_{\bar{N}}^k(s^k - s_*^k)_{\bar{N}} \end{pmatrix} + R(x^k)[X^k(s^k - s_*^k)]$$

where $\|R(x^k)\| = \mathcal{O}(\Delta_{\bar{N}}^k)$. This relation together with (61), (62), and statement (a) imply

$$\begin{aligned} \frac{\|X_{\bar{N}}^k(s^k - \tilde{s}^k)_{\bar{N}}\|}{\|x_{\bar{N}}^k\|} &\leq \frac{\|\tilde{P}_{\bar{N}}(x_{\bar{N}}^k)X_{\bar{N}}^k(s^k - s_*^k)_{\bar{N}}\|}{\|x_{\bar{N}}^k\|} + \|R(x^k)\| \frac{\|X^k(s^k - s_*^k)\|}{\|x_{\bar{N}}^k\|} \\ &\leq \|(s^k - s_*^k)_{\bar{N}}\| + \mathcal{O}(\Delta_{\bar{N}}^k) = \mathcal{O}(\|x^{k+1} - x^k\| + \Delta_{\bar{N}}^k), \end{aligned}$$

from which (b) follows.

We next prove (c). By (a), we have that $\liminf_{k \rightarrow \infty} \|X_{\bar{N}}^k s_{\bar{N}}^k\| / \|x_{\bar{N}}^k\| > 0$. Hence, by (b), we conclude that

$$(97) \quad \liminf_{k \rightarrow \infty} \|X_{\bar{N}}^k \tilde{s}_{\bar{N}}^k\| / \|x_{\bar{N}}^k\| > 0.$$

By Lemma 4.6, we have that $g_B^* \in \text{Range}(A_{\bar{B}}^T)$, or equivalently, that $\mathcal{D}_{\bar{B}}(x^*) \neq \emptyset$. Let $(\bar{y}, \bar{s})$ be an arbitrary element of $\mathcal{D}_{\bar{B}}(x^*)$. We have,

$$\begin{aligned} (g^*)^T(x^k - x^{k+1}) &= \frac{\beta}{\|X^k s^k\|} (g^*)^T (X^k)^2 s^k \\ &= \frac{\beta}{\|X^k s^k\|} \{ (g^*)^T (X^k)^2 \tilde{s}^k + (g^*)^T (X^k)^2 (s^k - \tilde{s}^k) \} \\ &= \frac{\beta}{\|X^k s^k\|} \{ \|X^k \tilde{s}^k\|^2 + \bar{s}^T (X^k)^2 (s^k - \tilde{s}^k) \} \\ &\geq \frac{\beta}{\|X^k s^k\|} \{ \|X_{\bar{N}}^k \tilde{s}_{\bar{N}}^k\|^2 - \|X_{\bar{N}}^k \bar{s}_{\bar{N}}\| \|X_{\bar{N}}^k (s^k - \tilde{s}^k)_{\bar{N}}\| \}. \end{aligned}$$

This inequality together with statements (a) and (b) imply that

$$\liminf_{k \rightarrow \infty} \frac{(g^*)^T(x^k - x^{k+1})}{\|x_{\bar{N}}^k\|} \geq \beta \liminf_{k \rightarrow \infty} \frac{\|X_{\bar{N}}^k \tilde{s}_{\bar{N}}^k\|^2}{\|X^k s^k\| \|x_{\bar{N}}^k\|},$$

which in view of (a) and (97) imply (c). $\square$

We are now in a position to state the main result of this paper. Its content is very similar to the one in Theorem 3.6. The main difference lies in the fact that Theorem 4.15 establishes the existence of a (unique) partition $(\bar{B}, \bar{N})$ satisfying the Local Assumption 3.2 while Theorem 3.6 assumes its existence.

THEOREM 4.15. *Let $\bar{B} \equiv \{1, \dots, n\} \setminus \bar{N}$, where $\bar{N}$ is a minimal element of Γ . Then the limit point x^* of the sequence $\{x^k\}$ generated by Algorithm AS is an optimal solution of problem (1) and the associated sequence of dual estimates $\{(y^k, s^k)\}$ converges to the optimal solution (y^*, s^*) of the dual problem (2) that solves the problem*

$$(98) \quad \begin{aligned} &\text{maximize} \quad \sum_{i \in \bar{N}} \log s_i \\ &\text{subject to} \quad A^T y + s = g^*, \\ &\quad \quad \quad s_{\bar{B}} = 0, \quad s_{\bar{N}} > 0. \end{aligned}$$

In particular, it follows that Γ has a unique minimal element, or equivalently, $\bar{N}$ is the smallest element of Γ (with respect to inclusion).

Proof. The result follows as an immediate consequence of Theorem 3.6 once we verify that the partition $(\bar{B}, \bar{N})$ satisfies Local Assumption 3.2. That $(\bar{B}, \bar{N})$ satisfies conditions (A) and (D) of Local Assumption 3.2 follows as a consequence of Theorem 4.13 and Lemma 4.5. We now show that $(\bar{B}, \bar{N})$ satisfies condition (B). Indeed, by Lemma 4.14(c), there exists $k_0 \geq 0$ such that $(g^*)^T(x^k - x^{k+1}) > 0$ for all $k \geq k_0$. This observation and the fact that $(g^*)^T(x^k - x^*) = \sum_{l=k}^{\infty} (g^*)^T(x^l - x^{l+1})$ imply that $(g^*)^T(x^k - x^*) \geq (g^*)^T(x^k - x^{k+1})$ for every $k \geq k_0$. In view of Lemma 4.14(c), we then conclude that

$$(99) \quad \liminf_{k \rightarrow \infty} \frac{(g^*)^T(x^k - x^*)}{\|x_N^k\|} \geq \liminf_{k \rightarrow \infty} \frac{(g^*)^T(x^k - x^{k+1})}{\|x_N^k\|} > 0,$$

which shows that $(\bar{B}, \bar{N})$ satisfies condition (B). We now show that $(\bar{B}, \bar{N})$ satisfies condition (C). First observe that

$$(100) \quad u^k - \tilde{u}^k = \frac{X^k(s^k - \tilde{s}^k)}{(g^*)^T(x^k - x^*)} = \frac{P(x^k)X^k(g^{k+1} - g^*)}{(g^*)^T(x^k - x^*)} = \frac{P(x^k)X^k(s^k - s_*^k)}{(g^*)^T(x^k - x^*)},$$

where $\{s_*^k\}$ is the sequence introduced in Lemma 4.8. Hence, by (62) and (99) and Lemma 4.14(a), we obtain

$$(101) \quad \begin{aligned} \limsup_{k \rightarrow \infty} \|u^k - \tilde{u}^k\| &\leq \limsup_{k \rightarrow \infty} \frac{\|X^k(s^k - s_*^k)\|}{(g^*)^T(x^k - x^*)} \leq \limsup_{k \rightarrow \infty} \frac{\mathcal{O}(\|X^k s^k\|)}{(g^*)^T(x^k - x^*)} \\ &\leq \limsup_{k \rightarrow \infty} \frac{\mathcal{O}(\|x_N^k\|)}{(g^*)^T(x^k - x^*)} < \infty. \end{aligned}$$

This shows that $\{u^k - \tilde{u}^k\}$ is bounded. That the condition $\lim_{k \rightarrow \infty} \|u_N^k - \tilde{u}_N^k\| = 0$ is satisfied follows as an immediate consequence of the first equality in (100), Lemma 4.14(b), (99). $\square$

Consider the following set associated with problem (1) and its dual (2):

$$(102) \quad \mathcal{S} \equiv \{(x, y, s) \mid x \in \mathcal{P}^+, (y, s) \in \mathcal{D}^+(x), x^T s = 0\}.$$

It is well known that $(x, y, s) \in \mathcal{S}$ if and only if x is an optimal solution of (1) and (x, y, s) is an optimal solution of (2). Define

$$\begin{aligned} B_{\mathcal{S}} &\equiv \{j \mid x_j > 0 \text{ for some } (x, y, s) \in \mathcal{S}\}; \\ N_{\mathcal{S}} &\equiv \{j \mid s_j > 0 \text{ for some } (x, y, s) \in \mathcal{S}\}; \\ J_{\mathcal{S}} &\equiv \{j \mid x_j = s_j = 0 \text{ for all } (x, y, s) \in \mathcal{S}\} = \{1, \dots, n\} \setminus (B_{\mathcal{S}} \cup N_{\mathcal{S}}). \end{aligned}$$

It is well known that $(B_{\mathcal{S}}, N_{\mathcal{S}}, J_{\mathcal{S}})$ forms a partition of $\{1, \dots, n\}$ and that there exists $(\bar{x}, \bar{y}, \bar{s}) \in \mathcal{S}$ such that $\bar{x}_{B_{\mathcal{S}}} > 0$ and $\bar{s}_{N_{\mathcal{S}}} > 0$.

THEOREM 4.16. *Let $\bar{N}$ denote the smallest element of Γ . Then $\bar{N} = N_{\mathcal{S}}$.*

Proof. Observe that the limit point (x^*, y^*, s^*) is an element of $\mathcal{S}$ such that $s_N^* > 0$. It follows from the definition of $N_{\mathcal{S}}$ that $\bar{N} \subseteq N_{\mathcal{S}}$. To prove the reverse inclusion, let $(\bar{x}, \bar{y}, \bar{s})$ be an element of $\mathcal{S}$ such that $\bar{x}_{B_{\mathcal{S}}} > 0$ and $\bar{s}_{N_{\mathcal{S}}} > 0$. Then, we have

$$f(x^k) - f(x^*) = f(x^k) - f(\bar{x}) \geq \nabla f(\bar{x})^T(x^k - \bar{x}) = \bar{s}^T(x^k - \bar{x}) = \bar{s}_{N_{\mathcal{S}}}^T x_{N_{\mathcal{S}}}^k \geq \bar{s}_j x_j^k,$$

and hence, $x_j^k = \mathcal{O}(f(x^k) - f(x^*))$ for every $j \in N_{\mathcal{S}}$. It follows from Lemma 4.14(a) that $x_i^k \sim x_j^k$ for every $i, j \in N_{\mathcal{S}}$. Since $\bar{N} \in \Gamma$, we have $\liminf_{k \rightarrow \infty} \Delta_{\bar{N}}^k = 0$ and, in view of the previous observation, this can only happen if $\bar{N} = N_{\mathcal{S}}$. $\square$

5. Concluding discussion. In this paper, we have established the global convergence of the affine scaling algorithm for convex QP without assuming any nondegeneracy assumption. We conclude the paper by pointing out a few problems which can be topics for future research.

The first problem is to remove the assumption that the set of optimal solutions of (1) is bounded. The only time we use this assumption in this paper is in showing the convergence of the sequence $\{x^k\}$, where we use a result of Jie Sun [17].

The second problem is to establish that $B_S = \bar{B}$. This fact would imply that when the QP problem (1) has a pair of primal and dual solutions satisfying strictly complementarity (that is, $J_S = \emptyset$) then the limit points x^* and (y^*, s^*) of Theorem 4.15 would satisfy the strictly complementarity condition that $x^* + s^* > 0$. This would generalize a result obtained by Dikin [6] (see also Vanderbei and Lagarias [26]) for primal nondegenerate LP problems and by Tsuchiya and Muramatsu [25] for degenerate LP problems which establishes that the limit points x^* and (y^*, s^*) of Theorem 4.15 always satisfy the strictly complementarity condition.

The last problem would be to show convergence of Algorithm AS with $\beta = 1$.

Appendix A. In this appendix we establish some technical results that are used in the paper.

LEMMA A.1. *If α_1 and α_2 are positive scalars, then*

$$|\log \alpha_1 - \log \alpha_2| \leq \frac{1}{\min\{\alpha_1, \alpha_2\}} |\alpha_1 - \alpha_2|.$$

The following lemma is equivalent to Lemma 6 of Dikin and Roos [8]. It is stated differently here for the sake of convenience.

LEMMA A.2. *Let $\Omega_0 \equiv \{w \in \mathbb{R}^q \mid e^T w = 1, \|w\| < 1\}$, and define the function $\phi^q : \Omega_0 \rightarrow \mathbb{R}$ as*

$$\phi^q(w) = q \log(1 - \|w\|) - \sum_{i=1}^q \log\left(1 - \frac{w_i}{\|w\|}\right) \quad \forall w \in \Omega_0.$$

Then, for every $w \in \Omega_0$, we have $\phi^q(w) \leq 0$, and $\phi^q(w) = 0$ if and only if $w = e/q$.

Proof. Let $\tilde{\Omega}_0 \equiv \{z \in \mathbb{R}^q \mid e^T z > 1, \|z\| = 1\}$ and define the function $g : \tilde{\Omega}_0 \rightarrow \mathbb{R}$ as

$$g(z) = q \log\left(1 - \frac{1}{e^T z}\right) - \sum_{i=1}^n \log(1 - z_i).$$

Given $w \in \Omega_0$, it is easy to see that $z \equiv w/\|w\| \in \tilde{\Omega}_0$ and $\phi^q(w) = g(z)$. It has been shown in Lemma 6 of Dikin and Roos [8] that for every $z \in \tilde{\Omega}_0$, we have $g(z) \leq 0$, and $g(z) = 0$ if and only if $z = e/\sqrt{q}$. It is now easy to verify that the result follows from the last two observations. $\square$

The proof of the following result can be found in Todd and Burrell [18], Lemma 2.3.

LEMMA A.3. *Let $\bar{c}, \bar{d}, \bar{x} \in \mathbb{R}^q$ be given, and consider the function*

$$\varphi(\alpha) = \bar{n} \log \bar{c}^T(\bar{x} + \alpha \bar{d}) - \sum_j \log(\bar{x}_j + \alpha \bar{d}_j),$$

defined over the interval $\bar{I}$ of α 's for which both logarithms are well defined. If $\bar{x}$ and $\bar{d}$ are not proportional, then φ has at most one stationary point, and if it has one, it is a

minimizer. In particular, φ is strictly quasi-convex, that is $\varphi(t) < \max\{\varphi(t_1), \varphi(t_2)\}$ for every $t_1 < t < t_2$ in $\bar{I}$.

Using the above two lemmas, we now prove the following result.

LEMMA A.4. *Let*

$$\Omega_1 \equiv \{(w, \lambda) \in \mathbb{R}^q \times \mathbb{R} \mid e^T w = 1, \lambda \|w\| < 1, \lambda \in [0, 1]\},$$

and define the function $\Phi^q : \Omega_1 \rightarrow \mathbb{R}$ as

$$\Phi^q(w, \lambda) = q \log(1 - \lambda \|w\|) - \sum_{i=1}^q \log \left(1 - \lambda \frac{w_i}{\|w\|} \right) \quad \forall (w, \lambda) \in \Omega_1.$$

Then, for every $(w, \lambda) \in \Omega_1$, we have $\Phi^q(w, \lambda) \leq 0$, and $\Phi^q(w, \lambda) = 0$ if and only if either $w = e/q$ or $\lambda = 0$.

Proof. Let $(w, \lambda) \in \Omega_1$ be given. If $w = e/q$ or $\lambda = 0$, then it is clear that $\Phi^q(w, \lambda) = 0$. Assume then that $w \neq e/q$ and $\lambda > 0$. We claim that $\Phi^q(w, \lambda) < 0$, from which the lemma follows. We divide the proof of this claim into two parts depending on whether $\|w\| \geq 1$ or $\|w\| < 1$.

Assume first that $\|w\| \geq 1$. In this case, we have $w_i/\|w\| \leq \|w\|$ for every i , with strict inequality holding for at least one index i . Using this observation and the fact that $\lambda > 0$, it is now easy to see that $\Phi^q(w, \lambda) < 0$.

Assume now that $\|w\| < 1$ and let $h(t) \equiv \Phi^q(w, t)$ for every $t \in [0, 1]$. By Lemma A.2 and the assumption that $w \neq e/q$, we have $h(1) < 0$. It is easy to see that the function $h(t)$ coincides with the function φ of Lemma A.3 on the interval $[0, 1]$ when $\bar{x} = w$, $\bar{x} = e$ and $\bar{d} = -w/\|w\|$. In particular, since $\bar{x}$ and $\bar{d}$ are not proportional, it follows that $h(t) < \max\{h(0), h(1)\} = 0$ for every $t \in (0, 1)$. Since $\lambda \in (0, 1]$, we obtain from the last two conclusions that $\Phi^q(w, \lambda) = h(\lambda) < 0$. $\square$

PROPOSITION A.5. *Assume that the sequences $\{\tilde{u}^k\} \subset \mathbb{R}^n$ and $\{\lambda_k\} \subset \mathbb{R}$ and the partition (B, N) of $\{1, \dots, n\}$ satisfy the following:*

- (a) *there exists $\delta \in (0, 1)$ such that $1 - \lambda_k \|\tilde{u}^k\| \geq \delta$ for every k sufficiently large;*
- (b) *$e^T \tilde{u}_N^k \rightarrow 1$ as $k \rightarrow \infty$;*
- (c) *$\|\tilde{u}_B^k\| \rightarrow 0$ as $k \rightarrow \infty$;*
- (d) *$\{\tilde{u}^k\}$ is bounded;*
- (e) *there exist constants $0 < \lambda_{\min} \leq \lambda_{\max} < 1$ such that $\lambda_k \in [\lambda_{\min}, \lambda_{\max}]$ for every k sufficiently large.*

Let

$$\phi_N(\tilde{u}^k, \lambda_k) \equiv |N| \log(1 - \lambda_k \|\tilde{u}^k\|) - \sum_{i \in N} \log \left(1 - \lambda_k \frac{\tilde{u}_i}{\|\tilde{u}^k\|} \right).$$

Then, for every k sufficiently large, $\phi_N(\tilde{u}^k, \lambda_k)$ is well defined and

$$(103) \quad \phi_N(\tilde{u}^k, \lambda_k) \leq \mathcal{O}(|e^T \tilde{u}_N^k - 1| + \|\tilde{u}_B^k\|^2).$$

Moreover, if $\lim_{k \rightarrow \infty} \phi_N(\tilde{u}^k, \gamma_k) = 0$ then $\tilde{u}_N^k \rightarrow e/|N|$.

Proof. It follows from (a) and the inequality

$$(104) \quad 1 - \lambda_k \frac{\tilde{u}_i^k}{\|\tilde{u}^k\|} \geq 1 - \lambda_k \geq 1 - \lambda_{\max} > 0$$

that $\phi_N(\tilde{u}^k, \lambda_k)$ is well defined for every k sufficiently large. Let $w_N^k \equiv \tilde{u}_N^k / e^T \tilde{u}_N^k$ for $k \geq 0$. Since $e^T w_N^k = 1$, we have $\|w_N^k\| \geq 1/\sqrt{|N|}$ for $k \geq 0$. In view of (b), this implies that

$$(105) \quad \liminf_{k \rightarrow 0} \|\tilde{u}_N^k\| \geq 1/\sqrt{|N|},$$

and hence,

$$(106) \quad \|\tilde{u}^k\| - \|\tilde{u}_N^k\| = \frac{\|\tilde{u}^k\|^2 - \|\tilde{u}_N^k\|^2}{\|\tilde{u}^k\| + \|\tilde{u}_N^k\|} \leq \frac{\|\tilde{u}_B^k\|^2}{2\|\tilde{u}_N^k\|} = \mathcal{O}(\|\tilde{u}_B^k\|^2).$$

Using this relation, (105), (b), (d), (e), and the definition of w_N^k , we obtain

$$(107) \quad \begin{aligned} \lambda_k \|w_N^k\| - \lambda_k \|\tilde{u}^k\| &= \lambda_k \left(\frac{\|\tilde{u}_N^k\|}{e^T \tilde{u}_N^k} - \|\tilde{u}^k\| \right) \\ &= \lambda_k \left(\frac{\|\tilde{u}_N^k\| (1 - e^T \tilde{u}_N^k)}{e^T \tilde{u}_N^k} + (\|\tilde{u}_N^k\| - \|\tilde{u}^k\|) \right) \\ &= \mathcal{O}(|e^T \tilde{u}_N^k - 1| + \|\tilde{u}_B^k\|^2) \end{aligned}$$

and

$$(108) \quad \begin{aligned} \lambda_k \frac{w_i^k}{\|w_N^k\|} - \lambda_k \frac{\tilde{u}_i^k}{\|\tilde{u}^k\|} &= \lambda_k \left(\frac{\tilde{u}_i^k}{\|\tilde{u}_N^k\|} - \frac{\tilde{u}_i^k}{\|\tilde{u}^k\|} \right) \\ &= \lambda_k \tilde{u}_i^k \frac{(\|\tilde{u}^k\| - \|\tilde{u}_N^k\|)}{\|\tilde{u}_N^k\| \|\tilde{u}^k\|} = \mathcal{O}(\|\tilde{u}_B^k\|^2). \end{aligned}$$

Relation (107) together with (a), (b), (c) imply that

$$(109) \quad \limsup_{k \rightarrow \infty} \lambda_k \|w_N^k\| \leq 1 - \delta,$$

and hence by (e), $(w_N^k, \lambda_k) \in \Omega_1$ for every k sufficiently large, where the set Ω_1 is defined in Lemma A.4. This observation together with (a), (104), (107), (108), Lemma A.1, and Lemma A.4 imply that

$$(110) \quad \begin{aligned} \phi_N(\tilde{u}^k, \lambda_k) &= \Phi^q(w_N^k, \lambda_k) + |N| [\log(1 - \lambda_k \|\tilde{u}^k\|) - \log(1 - \lambda_k \|w_N^k\|)] \\ &\quad + \sum_{i \in N} \left[\log \left(1 - \lambda_k \frac{\tilde{u}_i^k}{\|\tilde{u}^k\|} \right) - \log \left(1 - \lambda_k \frac{w_i^k}{\|w_N^k\|} \right) \right] \\ &= \Phi^q(w_N^k, \lambda_k) + \mathcal{O}(|e^T \tilde{u}_N^k - 1| + \|\tilde{u}_B^k\|^2) \\ &\leq \mathcal{O}(|e^T \tilde{u}_N^k - 1| + \|\tilde{u}_B^k\|^2), \end{aligned}$$

where Φ^q is the function defined in Lemma A.4 with $q \equiv |N|$. We thus have proved (103). To show the last part of the lemma, assume that $\lim_{k \rightarrow \infty} \phi_N(\tilde{u}^k, \lambda_k) = 0$. By (110), it follows that $\lim_{k \rightarrow \infty} \Phi^q(w_N^k, \lambda_k) = 0$. By (b), (d), (e), and the definition of $\{w_N^k\}$, it follows that $\{(w_N^k, \lambda_k)\}$ is a bounded sequence. Let $(\bar{w}_N, \bar{\lambda})$ be an accumulation point of $\{(w_N^k, \lambda_k)\}$. Using (e) and (109), we conclude that $(\bar{w}_N, \bar{\lambda}) \in \Omega_1$ and $\bar{\lambda} > 0$. By the continuity of Φ^q on Ω_1 , it then follows that $\Phi^q(\bar{w}_N, \bar{\lambda}) = 0$. Using Lemma A.4 and the fact that $\bar{\lambda} > 0$, we conclude that $\bar{w}_N = e/|N|$. Hence, $\lim_{k \rightarrow \infty} \tilde{u}_N^k = \lim_{k \rightarrow \infty} w_N^k = e/|N|$, where the first equality is due to (b) and the definition of w_N^k . $\square$

Appendix B. The objective of this section is to provide a proof of Lemma 3.1.

LEMMA B.1. *Let $h \in \mathbb{R}^n$ and a partition (B, N) of $\{1, \dots, n\}$ be given and define $p(x) = P(x)h$, $\tilde{p}_B(x_B) = \tilde{P}_B(x_B)h_B$ and $\tilde{p}_N(x_N) = \tilde{P}_N(x_N)h_N$. Then, the following statements hold:*

(a) *The vector $\delta_B(x) = p_B(x) - \tilde{p}_B(x_B)$ is the unique optimal solution of the problem*

$$(111) \quad \begin{aligned} & \text{minimize}_{\delta_B} \quad \frac{1}{2} \|\delta_B\|^2, \\ & \text{subject to} \quad A_B X_B \delta_B = -A_N X_N p_N(x); \end{aligned}$$

(b) *The pair $(p_B(x), \delta_N(x))$, where $\delta_N(x) \equiv p_N(x) - \tilde{p}_N(x_N)$, is the unique optimal solution of the problem*

$$(112) \quad \begin{aligned} & \text{minimize}_{(p_B, \delta_N)} \quad \frac{1}{2} \|p_B\|^2 + \frac{1}{2} \|\delta_N\|^2 - h_B^T p_B, \\ & \text{subject to} \quad A_B X_B p_B + A_N X_N \delta_N = -A_N X_N \tilde{p}_N(x_N). \end{aligned}$$

Proof. We first show (a). Since $p(x)$ is the optimal solution of (13), we know that

$$(113) \quad p(x) - h \in \text{Range}(XA^T), \quad AXp(x) = 0.$$

Since $\tilde{p}_B(x_B)$ is the optimal solution of (15), we have

$$(114) \quad \tilde{p}_B(x_B) - h_B \in \text{Range}(X_B A_B^T), \quad A_B X_B \tilde{p}_B(x_B) = 0.$$

From (113) and (114), it follows that

$$p_B(x) - \tilde{p}_B(x_B) \in \text{Range}(X_B A_B^T), \quad A_B X_B [p_B(x) - \tilde{p}_B(x_B)] = -A_N X_N p_N(x),$$

and this shows that $\delta_B(x) = p_B(x) - \tilde{p}_B(x_B)$ satisfies the optimality conditions for (111). Hence, (a) follows. We next show (b). Since $\tilde{p}_N(x_N)$ is the optimal solution of (14), we conclude

$$(115a) \quad \begin{pmatrix} 0 \\ \tilde{p}_N(x_N) - h_N \end{pmatrix} \in \text{Range}(XA^T),$$

$$(115b) \quad A_N X_N \tilde{p}_N(x_N) \in \text{Range}(A_B).$$

Hence, it follows from (113) and (115) that

$$(116a) \quad \begin{pmatrix} p_B(x) - h_B \\ \delta_N(x) \end{pmatrix} \in \text{Range}(XA^T),$$

$$(116b) \quad A_B X_B p_B(x) + A_N X_N \delta_N(x) = -A_N X_N \tilde{p}_N(x_N),$$

which shows that $(p_B(x), \delta_N(x))$ satisfies the optimality conditions for (112). $\square$

LEMMA B.2. *There exists a constant $K_0 > 0$ such that*

$$(117) \quad \|\delta_B(x)\| \leq K_0 \Delta_N(x) \|p_N(x)\| \quad \forall x \in \mathcal{P}^{++}.$$

Proof. By (113), we know that $A_N X_N p_N(x) \in \text{Range}(A_B)$. Hence, Lemma 4.1 implies the existence of a constant $K_0 > 0$ with the property that for every $x \in \mathcal{P}^{++}$ there exists a vector $u_B(x)$ satisfying

$$A_B u_B(x) = -A_N X_N p_N(x), \quad \|u_B(x)\| \leq K_0 \|X_N p_N(x)\|.$$

Thus, $X_B^{-1}u_B(x)$ is a feasible solution for (111). In view of Lemma B.1(a), this implies that for every $x \in \mathcal{P}^{++}$,

$$\begin{aligned} \|\delta_B(x)\| &\leq \|X_B^{-1}u_B(x)\| \leq \|x_B^{-1}\| \|u_B(x)\| \\ &\leq K_0 \|x_B^{-1}\| \|X_N p_N(x)\| \leq K_0 \Delta_N(x) \|p_N(x)\|. \quad \square \end{aligned}$$

LEMMA B.3. *There exist constants $K_1 > 0$ and $K_2 > 0$ with the property that, for every $x \in \mathcal{P}^{++}$ with $\Delta_N(x) \equiv \|x_B^{-1}\| \|x_N\|$ sufficiently small, there hold*

$$(118) \quad \|\delta_N(x)\| \leq K_1 [\Delta_N(x)^2 \|h_N\| + \Delta_N(x) \|h_B\|]$$

and

$$(119) \quad \|\delta_B(x)\| \leq K_2 [\Delta_N(x)^2 \|h_B\| + \Delta_N(x) \|h_N\|].$$

Proof. We first prove that (118) holds. From (115b) and (116b), we have that $A_N X_N \delta_N(x) \in \text{Range}(A_B)$. By Lemma 4.1, there exists a constant $K_0 > 0$ (same as in Lemma B.2) such that, for every $x \in \mathcal{P}^{++}$, there exists a vector $v_B(x)$ satisfying

$$(120) \quad A_B v_B(x) = -A_N X_N \delta_N(x), \quad \|v_B(x)\| \leq K_0 \|X_N \delta_N(x)\| \leq K_0 \|x_N\| \|\delta_N(x)\|.$$

Relation (116b) and (120) imply that $A_B[X_B p_B(x) - v_B(x)] = -A_N X_N \tilde{p}_N(x_N)$, and hence that the pair $(p_B(x) - X_B^{-1}v_B(x), 0)$ is feasible for (112). This together with Lemma B.1(b) imply that

$$\frac{1}{2} \|p_B(x)\|^2 + \frac{1}{2} \|\delta_N(x)\|^2 - h_B^T p_B(x) \leq \frac{1}{2} \|p_B(x) - X_B^{-1}v_B(x)\|^2 - h_B^T [p_B(x) - X_B^{-1}v_B(x)].$$

Rearranging this last expression and using relation (120) and the fact that $\|u\|^2 - \|v\|^2 \leq \|u - v\| \|u + v\|$ for any two vectors u and v , we obtain

$$\begin{aligned} \frac{1}{2} \|\delta_N(x)\|^2 &\leq \frac{1}{2} \{ \|p_B(x) - X_B^{-1}v_B(x)\|^2 - \|p_B(x)\|^2 \} + h_B^T [X_B^{-1}v_B(x)] \\ &\leq \frac{1}{2} \|2p_B(x) - X_B^{-1}v_B(x)\| \|X_B^{-1}v_B(x)\| + \|h_B\| \|X_B^{-1}v_B(x)\| \\ &\leq \|x_B^{-1}\| \|v_B(x)\| \left\{ \frac{1}{2} \|2p_B(x) - X_B^{-1}v_B(x)\| + \|h_B\| \right\} \\ &\leq K_0 \Delta_N(x) \|\delta_N(x)\| \left\{ \|p_B(x)\| + \frac{1}{2} \|x_B^{-1}\| \|v_B(x)\| + \|h_B\| \right\}, \end{aligned}$$

from which it follows that

$$\|\delta_N(x)\| \leq 2K_0 \Delta_N(x) \left\{ \|p_B(x)\| + \frac{1}{2} \|x_B^{-1}\| \|v_B(x)\| + \|h_B\| \right\}.$$

We now bound the term within the parentheses in the last expression. Using (120), Lemma B.2, and the fact that $\|\tilde{p}_B(x_B)\| \leq \|h_B\|$ and $\|\tilde{p}_N(x_N)\| \leq \|h_N\|$, we obtain

$$\begin{aligned} \|p_B(x)\| + \frac{1}{2} \|x_B^{-1}\| \|v_B(x)\| + \|h_B\| \\ \leq \|\delta_B(x)\| + \|\tilde{p}_B(x_B)\| + \frac{1}{2} K_0 \|x_B^{-1}\| \|X_N\| \|\delta_N(x)\| + \|h_B\| \end{aligned}$$

$$\begin{aligned}
&\leq K_0 \Delta_N(x) \|p_N(x)\| + \frac{1}{2} K_0 \Delta_N(x) \|\delta_N(x)\| + 2\|h_B\| \\
&\leq K_0 \Delta_N(x) \left[\|p_N(x)\| + \frac{1}{2} \|\delta_N(x)\| \right] + 2\|h_B\| \\
&\leq K_0 \Delta_N(x) \left[\|\tilde{p}_N(x)\| + \frac{3}{2} \|\delta_N(x)\| \right] + 2\|h_B\| \\
&\leq K_0 \Delta_N(x) \left[\|h_N\| + \frac{3}{2} \|\delta_N(x)\| \right] + 2\|h_B\|.
\end{aligned}$$

Combining the last two inequalities and rearranging, we have

$$\begin{aligned}
\|\delta_N(x)\| &\leq \frac{1}{1 - 3K_0^2 \Delta_N(x)^2} [2K_0^2 \Delta_N(x)^2 \|h_N\| + 4K_0 \Delta_N(x) \|h_B\|] \\
&\leq K_1 [\Delta_N(x)^2 \|h_N\| + \Delta_N(x) \|h_B\|],
\end{aligned}$$

for every $x \in \mathcal{P}^{++}$ such that $\Delta_N(x) \leq 1/\sqrt{6K_0}$, where $K_1 \equiv 4 \max\{K_0^2, 2K_0\}$. We next show that (119) holds. Using Lemma B.2 and the fact that $\|\tilde{p}_N(x_N)\| \leq \|h_N\|$, we obtain

$$\begin{aligned}
\|\delta_B(x)\| &\leq K_0 \Delta_N(x) \|p_N(x)\| \\
&\leq K_0 \Delta_N(x) [\|\tilde{p}_N(x_N)\| + \|\delta_N(x)\|] \\
&\leq K_0 \Delta_N(x) [\|h_N\| + K_1 \Delta_N(x)^2 \|h_N\| + K_1 \Delta_N(x) \|h_B\|] \\
&\leq K_2 [\Delta_N(x)^2 \|h_B\| + \Delta_N(x) \|h_N\|],
\end{aligned}$$

for every $x \in \mathcal{P}^{++}$ such that $\Delta_N(x) \leq 1/\sqrt{K_1}$, where $K_2 \equiv \max\{2K_0, K_0 K_1\}$. $\square$

We are now ready to give the proof of Lemma 3.1.

Proof of Lemma 3.1. To show (a), let $x \in \mathcal{P}^{++}$ be given. The symmetry of the matrix $R(x)$ follows trivially from the fact that $P(x)$, $\tilde{P}_B(x_B)$ and $\tilde{P}_N(x_N)$ are symmetric matrices. From (115b) and (116b), we have that $A_N X_N \delta_N(x) \in \text{Range}(A_B)$. Hence, it follows from (14) that $\tilde{P}_N(x_N) \delta_N(x) = \delta_N(x)$, or equivalently,

$$\tilde{P}_N(x_N) R_{NB}(x) h_B + \tilde{P}_N(x_N) R_{NN}(x) h_N = R_{NB}(x) h_B + R_{NN}(x) h_N.$$

Since the last relation holds for every vector $h = (h_B, h_N)$, we conclude that

$$\tilde{P}_N(x_N) R_{NB}(x) = R_{NB}(x), \text{ and } \tilde{P}_N(x_N) R_{NN}(x) = R_{NN}(x).$$

This shows the first relation in (17). Using the fact that $\tilde{P}_N(x_N) R_{NN}(x) = R_{NN}(x)$ and the fact that $R_{NN}(x)$ and $\tilde{P}_N(x_N)$ are symmetric matrices, we obtain

$$\begin{aligned}
\tilde{P}_N(x_N) R_{NN}(x) \tilde{P}_N(x_N) &= R_{NN}(x) \tilde{P}_N(x_N) = \left(\tilde{P}_N(x_N) R_{NN}(x) \right)^T \\
&= R_{NN}(x)^T = R_{NN}(x),
\end{aligned}$$

which shows the second relation in (17). We now show (b). Relation (118) implies that

$$\|\delta_B(x)\| = \|R_{BB}(x) h_B + R_{BN}(x) h_N\| \leq K_1 (\Delta_N(x)^2 \|h_B\| + \Delta_N(x) \|h_N\|),$$

for every $x \in \mathcal{P}^{++}$ with $\Delta_N(x)$ sufficiently small. Since this relation holds for every $h = (h_B, h_N)$, we conclude that $\|R_{BB}(x)\| \leq K_1 \Delta_N(x)^2$ and $\|R_{BN}(x)\| \leq K_1 \Delta_N(x)$,

for every $x \in \mathcal{P}^{++}$ with $\Delta_N(x)$ sufficiently small. Similarly, using (119), it is easy to see that $\|R_{NN}(x)\| \leq K_2 \Delta_N(x)^2$, for every $x \in \mathcal{P}^{++}$ with $\Delta_N(x)$ sufficiently small. Hence (b) follows if we let $K_0 \equiv \max\{K_1, K_2\}$.

Acknowledgement. The second author thanks Dr. Masakazu Muramatsu of the Department of Mechanical Engineering of Sophia University, Tokyo, Japan, for his useful insights and comments on this paper.

REFERENCES

- [1] I. ADLER AND R. D. C. MONTEIRO, *Limiting behavior of the affine scaling continuous trajectories for linear programming problems*, Math. Programming, 50 (1991), pp. 29–51.
- [2] E. R. BARNES, *A variation on Karmarkar's algorithm for solving linear programming problems*, Math. Programming, 36 (1986), pp. 174–182.
- [3] D. A. BAYER AND J. C. LAGARIAS, *The nonlinear geometry of linear programming, part I: Affine and projective scaling trajectories*, Trans. Amer. Math. Soc., 314 (1989), pp. 499–526.
- [4] J. BONNANS AND M. BOUHTOU, *The trust region affine interior point algorithm for convex and nonconvex quadratic programming*, RAIRO Rech. Opér., 29 (1995), pp. 195–217.
- [5] I. I. DIKIN, *Iterative solution of problems of linear and quadratic programming*, Dok. Akad. Nauk SSSR, 174 (1967), pp. 747–748 (in Russian); Soviet Math. Dokl., 8 (1967), pp. 674–675 (in English).
- [6] I. I. DIKIN, *On the convergence of an iterative process*, Upravlyaemye Sistemy, 12 (1974), pp. 54–60 (in Russian).
- [7] I. I. DIKIN, *The Convergence of Dual Variables*, Tech. report, Siberian Energy Institute, Irkutsk, Russia, December 1991.
- [8] I. I. DIKIN AND C. ROOS, *Convergence of the dual variables for the primal affine scaling method with unit steps in the homogeneous case*, Reports of the Faculty of Technical Mathematics and Informatics 94–96, Technical Mathematics and Informatics Department, Delft University of Technology, Delft, the Netherlands, 1994.
- [9] I. I. DIKIN AND V. I. ZORKALTSEV, *Iterative Solutions of Mathematical Programming Problems*, Nauka, Novosibirsk, 1980.
- [10] C. C. GONZAGA, *Convergence of the Large Step Primal Affine-Scaling Algorithm for Primal Nondegenerate Linear Programs*, Tech. report ES-230/90, Department of Systems Engineering and Computer Science, COPPE Federal University of Rio de Janeiro, 21941 Rio de Janeiro, RJ, Brazil, September 1990.
- [11] C. C. GONZAGA AND L. A. CARLOS, *A Primal Affine-Scaling Algorithm for Linearly Constrained Convex Programs*, Tech. report ES-238/90, Department of Systems Engineering and Computer Science, COPPE Federal University of Rio de Janeiro, 21941 Rio de Janeiro, RJ, Brazil, December 1990.
- [12] N. K. KARMAKAR, *A new polynomial-time algorithm for linear programming*, Combinatorica, 4 (1984), pp. 373–395.
- [13] N. MEGIDDO AND M. SHUB, *Boundary behavior of interior point algorithms in linear programming*, Math. Oper. Res., 14 (1989), pp. 97–114.
- [14] R. D. C. MONTEIRO, *Convergence and boundary behavior of the projective scaling trajectories for linear programming*, Math. Oper. Res., 16 (1991), pp. 842–858.
- [15] R. D. C. MONTEIRO, T. TSUCHIYA, AND Y. WANG, *A simplified global convergence proof of the affine scaling algorithm*, Ann. Oper. Res., 47 (1993), pp. 443–482.
- [16] R. SAIGAL, *A simple proof of a primal affine scaling method*, Ann. Oper. Res., 62 (1996), pp. 303–324.
- [17] J. SUN, *A convergence proof for an affine-scaling algorithm for convex quadratic programming without nondegeneracy assumptions*, Math. Programming, 60 (1993), pp. 69–79.
- [18] M. J. TODD AND B. P. BURRELL, *An extension of Karmarkar's algorithm for linear programming using dual variables*, Algorithmica, 1 (1986), pp. 409–424.
- [19] P. TSENG AND Z. Q. LUO, *On the convergence of the affine-scaling algorithm*, Math. Programming, 56 (1992), pp. 301–319.
- [20] T. TSUCHIYA, *Global convergence of the affine scaling algorithm for degenerate strictly convex quadratic programming problems*, talk presented at TIMS/ORSA Joint National Meeting, Chicago, IL, May 1993.
- [21] T. TSUCHIYA, *Global convergence of the affine-scaling methods for degenerate linear programming problems*, Math. Programming, 52 (1991), pp. 377–404.

- [22] T. TSUCHIYA, *Global convergence property of the affine scaling method for primal degenerate linear programming problems*, Math. Oper. Res., 17 (1992), pp. 527–557.
- [23] T. TSUCHIYA, *Global convergence of the affine scaling algorithm for primal degenerate strictly convex quadratic programming problems*, Ann. Oper. Res., 47 (1993), pp. 509–539.
- [24] T. TSUCHIYA AND R. D. C. MONTEIRO, *Superlinear convergence of the affine scaling algorithm*, Math. Programming, 75 (1996), pp. 77–110.
- [25] T. TSUCHIYA AND M. MURAMATSU, *Global convergence of a long-step affine scaling algorithm for degenerate linear programming problems*, SIAM J. Optim., 5 (1995), pp. 525–551.
- [26] R. J. VANDERBEI AND J. C. LAGARIAS, *I. I. Dikin's convergence result for the affine-scaling algorithm*, in Mathematical Developments Arising from Linear Programming: Proceedings of a Joint Summer Research Conference held at Bowdoin College, Brunswick, Maine, June/July 1988, J. C. Lagarias and M. J. Todd, eds., Contemp. Math. 114, AMS, Providence, RI, 1990, pp. 109–119.
- [27] R. J. VANDERBEI, M. S. MEKTON, AND B. A. FREEDMAN, *A modification of Karmarkar's linear programming algorithm*, Algorithmica, 1 (1986), pp. 395–407.
- [28] C. WITZGALL, P. T. BOGGS, AND P. D. DOMICH, *On the convergence behavior of trajectories for linear programming*, in Mathematical Developments Arising from Linear Programming: Proceedings of a Joint Summer Research Conference held at Bowdoin College, Brunswick, ME, June/July 1988, J. C. Lagarias and M. J. Todd, eds., Contemp. Math. 114, AMS, Providence, RI, 1990, pp. 161–187.
- [29] Y. YE, *An extension of Karmarkar's projective algorithm and the trust region method for quadratic programming*, in Progress in Mathematical Programming: Interior Point and Related Methods, N. Megiddo, ed., Springer-Verlag, Berlin, 1989, pp. 49–63.
- [30] Y. YE, *On an affine scaling algorithm for nonconvex quadratic programming*, Math. Programming, 56 (1992), pp. 285–300.
- [31] Y. YE AND E. TSE, *An extension of Karmarkar's projective algorithm for convex quadratic programming*, Math. Programming, 44 (1989), pp. 157–180.

SUPERLINEAR CONVERGENCE OF A SYMMETRIC PRIMAL-DUAL PATH FOLLOWING ALGORITHM FOR SEMIDEFINITE PROGRAMMING*

ZHI-QUAN LUO[†], JOS F. STURM[‡], AND SHUZHONG ZHANG[‡]

Abstract. This paper establishes the superlinear convergence of a symmetric primal-dual path following algorithm for semidefinite programming (SDP) under the assumptions that the semidefinite program has a strictly complementary primal-dual optimal solution and that the size of the central path neighborhood tends to zero. The interior point algorithm considered here closely resembles the Mizuno–Todd–Ye predictor-corrector method for linear programming which is known to be quadratically convergent. It is shown that when the iterates are well centered, the duality gap is reduced superlinearly after each predictor step. Indeed, if each predictor step is succeeded by r consecutive corrector steps then the predictor reduces the duality gap superlinearly with order $2/(1 + 2^{-r})$. The proof relies on a careful analysis of the central path for SDP. It is shown that under the strict complementarity assumption, the primal-dual central path converges to the analytic center of the primal-dual optimal solution set, and the distance from any point on the central path to this analytic center is bounded by the duality gap.

Key words. semidefinite programming, central path, path following, superlinear convergence

AMS subject classifications. 90C25, 90C26, 90C60

PII. S1052623496299187

1. Introduction. Recently, there have been many interior point algorithms developed for SDP; see, for example, [1, 2, 4, 8, 12, 14, 15, 17, 21]. These algorithms differ in their choices of scaling matrix, the size of the central path neighborhoods, and stepsize rules, among others. In particular, the algorithms of Kojima, Shindoh, and Hara [8] and Nesterov and Todd [14, 15] are based on the primal-dual scaling and they both can be viewed as extensions of the predictor-corrector method for linear programming [11]. It has been shown [7, 9, 14, 15, 17, 21] that these algorithms for SDP retain many important properties of the interior point algorithms for linear programming including polynomial complexity. For an overview of SDP and its applications, we refer to Vandenberghe and Boyd [19].

However, there exists considerable difficulty in extending one key property of the predictor-corrector method for linear programming to the interior point algorithms for SDP. This is the property of quadratic convergence of the duality gap (see [20] for a proof of the linear complementarity problem (LCP) case). In some sense, the need for superlinear convergence in solving SDP is more pronounced than that for the linear programming case. This is because for SDP there cannot exist any finite termination procedures as in the case of linear programming. Indeed, the recent papers of Kojima, Shida, and Shindoh [7] and Potra and Sheng [16] are both focused on the issue of superlinear convergence for solving SDP. In particular, the latter reference provided

*Received by the editors February 16, 1996; accepted for publication (in revised form) December 17, 1996.

<http://www.siam.org/journals/siopt/8-1/29918.html>

[†]Room 225/CRL, Department of Electrical and Computer Engineering, McMaster University, Hamilton, Ontario, L8S 4L7, Canada (luozq@mcmaster.ca). The research of this author was supported by Natural Sciences and Engineering Research Council of Canada grant OPG0090391 and was performed during a research leave to the Econometric Institute, Erasmus University Rotterdam.

[‡]Econometric Institute, Erasmus University Rotterdam, P.O. Box 1738, 3000 DR Rotterdam, The Netherlands (sturm@few.eur.nl, zhang@few.eur.nl).

a sufficient condition for the superlinear convergence of an infeasible path following algorithm, while the former reference [7] established the superlinear convergence of their algorithm [8] under certain key assumptions. These assumptions are (1) SDP is nondegenerate in the sense that the Jacobian matrix of its Karush–Kuhn–Tucker (KKT) system is nonsingular; (2) SDP has a strictly complementary optimal solution; (3) the iterates converge tangentially to the central path in the sense that the size of the central path neighborhood in which the iterates reside must tend to zero. Among these three assumptions for superlinear convergence, (2) is inevitable since it is needed even in the case of LCP (see [20]). Assumption (3) is needed to ensure the duality gap is reduced superlinearly after each predictor step for *all* points in the central path neighborhood. In reference [7], an example was given which showed that, without the tangential convergence assumption, the duality gap is reduced only linearly after one predictor step for certain points in the central path neighborhood.

Our goal in this paper is to establish the superlinear convergence of a symmetric path following algorithm for SDP under assumptions (2) and (3) only (i.e., without the nondegeneracy assumption). In particular, we consider the primal-dual path following algorithm of Nesterov and Todd [14, 15] (later discovered independently by Sturm and Zhang [17] using a V -space notion). In this paper we adopt the framework of [17] since it greatly facilitates the subsequent analysis. We show that this symmetric primal-dual path following algorithm has an order of convergence that is asymptotically quadratic (i.e., subquadratic). Indeed, for any given constant positive integer r , the algorithm can be set so that the duality gap decreases superlinearly with order $2/(1 + 2^{-r})$ after one predictor (affine scaling) step followed by (at most) r corrector steps. The cornerstone in our bid to establish this superlinear convergence result is a bound on the distance from any point on the central path to the optimal solution set (see section 3). Specifically, it is shown that, under the strict complementarity assumption, the primal-dual central path converges to the analytic center of the optimal solution set, and that the distance to this analytic center from any point on the central path can be bounded above by the duality gap. These properties of the central path are algorithm-independent and are likely to be useful in the analysis of other interior point algorithms for SDP.

The organization of this paper is as follows. At the end of this section, we describe some basic notation to be used in this paper. In section 2, we will discuss some fundamental background notions, and we will make two assumptions concerning the solution set of the SDP. In section 3 we will analyze the limiting behavior of the primal-dual central path. In section 4, the notion of V -space for SDP is reviewed and a path following algorithm in the spirit of [17] is introduced. The superlinear convergence of this algorithm is established in section 5. Finally, some concluding remarks are given in section 6.

Notation. The space of symmetric $n \times n$ matrices will be denoted $\mathcal{S}$. Given matrices X and Y in $\mathfrak{R}^{m_1 \times m_2}$, the standard inner product is defined by

$$X \bullet Y = \operatorname{tr} X^T Y,$$

where $\operatorname{tr}(\cdot)$ denotes the trace of a matrix. The notation $X \perp Y$ denotes orthogonality in the sense that $X \bullet Y = 0$. The Euclidean norm and its associated operator norm, viz. the spectral norm, are both denoted by $\|\cdot\|$. The Frobenius norm of X is $\|X\|_F = \sqrt{X \bullet X}$. If $X \in \mathcal{S}$ is positive (semi-) definite, we write $(X \succeq 0)$ $X \succ 0$. The cone of positive semidefinite matrices is denoted by $\mathcal{S}_+$ and the cone of positive definite matrices is $\mathcal{S}_{++}$. The identity matrix is denoted by I . We use the standard Landau

notation (“big O ” and “small o ”). In particular, if $\{u(\mu) : \mu > 0\}$ and $\{w(\mu) : \mu > 0\}$ are real sequences with $w(\mu) > 0$, then $u(\mu) = O(w(\mu))$ means that $u(\mu)/w(\mu)$ is bounded, independent of μ ; $u(\mu) = o(w(\mu))$ means that $\lim_{\mu \rightarrow 0} u(\mu)/w(\mu) = 0$; $u(\mu) \sim w(\mu)$ means that $\lim_{\mu \rightarrow 0} u(\mu)/w(\mu) = 1$; $u(\mu) = \Theta(w(\mu))$ whenever $u(\mu)/w(\mu)$ and $w(\mu)/u(\mu)$ are both bounded. For a symmetric positive definite matrix, we use “ O ” and “ Θ ” to denote the order of all its eigenvalues. Hence, for $U(\mu) \in \mathcal{S}_{++}$, the notation $U(\mu) = \Theta(w(\mu))$ signifies the existence of $\Gamma > 0$ such that

$$\frac{1}{\Gamma}I \preceq \frac{1}{w(\mu)}U(\mu) \preceq \Gamma I \quad \text{for all } \mu > 0.$$

Therefore, the condition $U(\mu) = \Theta(w(\mu))$ implies that $\|U(\mu)\| = \Theta(w(\mu))$ and the diagonal entries

$$U_{11}(\mu), U_{22}(\mu), \dots, U_{nn}(\mu)$$

are all $\Theta(w(\mu))$ too.

2. Problem formulation. An SDP problem is given as

$$\begin{aligned} & \text{minimize} && C \bullet X \\ \text{(P)} \quad & \text{subject to} && A^{(i)} \bullet X = b_i \quad \text{for } i = 1, 2, \dots, m, \\ & && X \succeq 0 \end{aligned}$$

where $C \in \mathcal{S}$, $A^{(1)}, A^{(2)}, \dots, A^{(m)} \in \mathcal{S}$ and $b \in \mathbb{R}^m$. The decision variable is $X \in \mathcal{S}$. The corresponding dual program can be formulated as

$$\begin{aligned} & \text{maximize} && b^T y \\ \text{(D)} \quad & \text{subject to} && \sum_{i=1}^m y_i A^{(i)} + Z = C, \\ & && Z \succeq 0. \end{aligned}$$

Denote the feasible sets of (P) and (D) by $\mathcal{F}_P$ and $\mathcal{F}_D$, respectively, i.e.,

$$\mathcal{F}_P := \{X \in \mathcal{S} : A^{(i)} \bullet X = b_i, i = 1, 2, \dots, m, X \succeq 0\}$$

and

$$\mathcal{F}_D := \{Z \in \mathcal{S} : Z = C - \sum_{i=1}^m y_i A^{(i)} \text{ for some } y \in \mathbb{R}^m, Z \succeq 0\}.$$

We make the following assumptions throughout this paper.

ASSUMPTION 2.1. *There exist positive definite solutions $X \in \mathcal{F}_P$ and $Z \in \mathcal{F}_D$ for (P) and (D), respectively.*

ASSUMPTION 2.2. *There exists a pair of strictly complementary primal-dual optimal solutions for (P) and (D). Specifically, there exists $(X^*, Z^*) \in \mathcal{F}_P \times \mathcal{F}_D$ such that*

$$\begin{cases} X^* Z^* = 0, \\ X^* + Z^* \succ 0. \end{cases}$$

Since $X^*Z^* = Z^*X^* = 0$, we can diagonalize X^* and Z^* simultaneously. Therefore, by applying an orthonormal transformation to the problem data if necessary, we can assume without loss of generality that X^* , Z^* are both diagonal and of the form

$$(2.1) \quad X^* = \begin{bmatrix} \Lambda_B & 0 \\ 0 & 0 \end{bmatrix}, \quad Z^* = \begin{bmatrix} 0 & 0 \\ 0 & \Lambda_N \end{bmatrix},$$

where $\Lambda_B := \text{diag}(\lambda_1, \dots, \lambda_K)$, $\Lambda_N := \text{diag}(\lambda_{K+1}, \dots, \lambda_n)$ for some integer $0 \leq K \leq n$ and some positive scalars $\lambda_i > 0$, $i = 1, 2, \dots, n$. Here the subscripts B and N signify the “basic” and “nonbasic” subspaces (following the terminology of linear programming). Throughout this paper, the decomposition of any $n \times n$ matrix X is always made with respect to the above partition B and N . In fact, we shall adhere to the following notation throughout:

$$X = \begin{bmatrix} X_B & X_U \\ X_U^T & X_N \end{bmatrix},$$

so X_U will always denote the off-diagonal block of X with size $K \times (n - K)$, etc.

Notice that $X \in \mathcal{F}_P$ is an optimal solution to (P) if and only if $XZ^* = 0$. Hence, by Assumption 2.2, the primal optimal solution set can be written as

$$\mathcal{F}_P^* := \{X \in \mathcal{F}_P : X_U = 0 \text{ and } X_N = 0\}.$$

Analogously, the dual optimal solution set is given by

$$\mathcal{F}_D^* := \{Z \in \mathcal{F}_D : Z_U = 0 \text{ and } Z_B = 0\}.$$

Given $\mu \in \mathfrak{R}_{++}$, the pair $(X, Z) \in \mathcal{F}_P \times \mathcal{F}_D$ is said to be the μ -center $(X(\mu), Z(\mu))$ if and only if

$$(2.2) \quad XZ = \mu I.$$

We refer to [8, 18] for a proof of the existence and uniqueness of μ -centers. The *central path* of the problem (P) is the curve

$$\{(X(\mu), Z(\mu)) : \mu > 0\}.$$

The logarithmic barrier function $\log \det X_B$ is strictly concave on the relative interior of the primal optimal solution set

$$\{X \in \mathcal{F}_P^* : X_B \succ 0\}.$$

Moreover, Assumption 2.1 implies that $\mathcal{F}_P^*$ and $\mathcal{F}_D^*$ are bounded. The unique maximizer X^a of $\log \det X_B$ on the relative interior of $\mathcal{F}_P^*$ is called the *analytic center* of $\mathcal{F}_P^*$. It is characterized by the KKT system

$$(2.3) \quad \begin{cases} X_B^a Z_B = I, \\ \sum_{i=1}^m y_i A_B^{(i)} + Z_B = 0, \\ X^a \in \mathcal{F}_P^* \text{ and } Z_B \succ 0. \end{cases}$$

In a similar fashion, we define the analytic center of $\mathcal{F}_D^*$ as the maximizer of the logarithmic barrier $\log \det Z_N$ on the relative interior of $\mathcal{F}_D^*$; Z^a is characterized by the KKT system

$$\begin{cases} X_N Z_N^a = I, \\ A^{(i)} \bullet X = 0, & i = 1, 2, \dots, m, \\ Z^a \in \mathcal{F}_D^* \text{ and } X_N \succ 0. \end{cases}$$

3. Properties of the central path. The notion of central path plays a fundamental role in the development of interior point methods for linear programming. In this section, we shall study the analytic properties of the central path in the context of semidefinite programming. These properties will be used in section 5 where we perform convergence analysis of a predictor-corrector algorithm for SDP.

For linear programming (i.e., $A^{(i)}$'s and C are diagonal), it is known that the central path curve converges: $(X(\mu), Z(\mu)) \rightarrow (X^a, Z^a)$, as $\mu \rightarrow 0$, with (X^a, Z^a) being the analytic center of the primal and dual optimal solution sets $\mathcal{F}_P^*$ and $\mathcal{F}_D^*$, respectively [10]. Another important property of the central path in the context of linear programming is that it never converges *tangentially* to the optimal face [13]. This means that for any point on the central path, the distance to the end of the central path is of the same order as the distance to the optimal face, viz. $O(\mu)$. The aim of this section is to establish a similar property of the central path for semidefinite programming. More specifically, we shall prove that

$$\|X(\mu) - X^a\| + \|Z(\mu) - Z^a\| = O(\mu).$$

We begin with the following lemma which shows that the set

$$\{(X(\mu), Z(\mu)) : 0 < \mu < 1\}$$

is bounded.

LEMMA 3.1. *For any $\mu > 0$ there holds*

$$\|X(\mu)\| + \|Z(\mu)\| = O(1 + \mu).$$

Proof. Since $A^{(i)} \bullet (X(\mu) - X(1)) = 0$ for $i = 1, \dots, m$, and $Z(\mu) - Z(1) \in \text{Span}\{A^{(i)}, i = 1, \dots, m\}$, it follows that $(X(\mu) - X(1)) \perp (Z(\mu) - Z(1))$. Using this property, we obtain

$$\begin{aligned} n\mu + n &= X(\mu) \bullet Z(\mu) + X(1) \bullet Z(1) \\ &= X(1) \bullet Z(\mu) + Z(1) \bullet X(\mu). \end{aligned}$$

Since $X(1) \succ 0$ and $Z(1) \succ 0$, we have

$$\|X(\mu)\| + \|Z(\mu)\| = O(X(1) \bullet Z(\mu) + Z(1) \bullet X(\mu)) = O(1 + \mu). \quad \square$$

It follows from Lemma 3.1 that the central path has a limit point. We will now show that any limit point of the central path $\{(X(\mu), Z(\mu))\}$ is a strictly complementary optimal primal-dual pair.

LEMMA 3.2. *For any $\mu \in (0, 1)$ there holds*

$$\begin{aligned} X_B(\mu) &= \Theta(1), & X_N(\mu) &= \Theta(\mu), \\ Z_B(\mu) &= \Theta(\mu), & Z_N(\mu) &= \Theta(1). \end{aligned}$$

Hence, any limit point of $\{(X(\mu), Z(\mu))\}$ as $\mu \rightarrow 0$ is a pair of strictly complementary primal-dual optimal solutions of (P) and (D).

Proof. Let $0 < \mu < 1$. For notational convenience, we will use X and Z to denote the matrices $X(\mu)$ and $Z(\mu)$. Let (X^*, Z^*) be the pair of strictly complementary

primal-dual optimal solutions postulated by Assumption 2.2. Since $(X - X^*) \perp (Z - Z^*)$, we have

$$\begin{aligned}
0 &= (X - X^*) \bullet (Z - Z^*) \\
&= X \bullet Z - X^* \bullet Z - X \bullet Z^* \\
&= \text{tr}(\mu I - X^* Z - X Z^*) \\
&= n\mu - \sum_{i=1}^K \lambda_i Z_{ii} - \sum_{i=K+1}^n \lambda_i X_{ii},
\end{aligned}$$

where the last step follows from (2.1). Since $\lambda_i > 0$ for all i and $X_{ii} \geq 0$ and $Z_{ii} \geq 0$ (by the positive semidefiniteness of X and Z), we obtain

$$\begin{cases} Z_{ii} = O(\mu), & i = 1, \dots, K, \\ X_{ii} = O(\mu), & i = K+1, \dots, n. \end{cases}$$

Since $X \succeq 0$, $Z \succeq 0$, it follows that

$$(3.1) \quad X_N = O(\mu), \quad Z_B = O(\mu).$$

From $X \succ 0$ and $Z \succ 0$ we obtain

$$X_N - X_U^T X_B^{-1} X_U \succ 0, \quad Z_B - Z_U Z_N^{-1} Z_U^T \succ 0.$$

Now consider the identities

$$\begin{aligned}
\log \det X &= \log \det X_B + \log \det(X_N - X_U^T X_B^{-1} X_U), \\
\log \det Z &= \log \det Z_N + \log \det(Z_B - Z_U Z_N^{-1} Z_U^T).
\end{aligned}$$

Since $\det X \det Z = \det(\mu I) = \mu^n$, it follows that $\log \det X + \log \det Z = n \log \mu$ and

$$\begin{aligned}
0 &= \log \det X_B + \log \det \left(\frac{1}{\mu} (X_N - X_U^T X_B^{-1} X_U) \right) \\
&\quad + \log \det Z_N + \log \det \left(\frac{1}{\mu} (Z_B - Z_U Z_N^{-1} Z_U^T) \right).
\end{aligned}$$

By the estimates (3.1) and using Lemma 3.1, we see that

$$X_B = O(1), \quad \frac{1}{\mu} (X_N - X_U^T X_B^{-1} X_U) = O(1),$$

$$Z_N = O(1), \quad \frac{1}{\mu} (Z_B - Z_U Z_N^{-1} Z_U^T) = O(1).$$

Therefore, each of the four logarithm terms in the preceding equation are bounded from above as $\mu \rightarrow 0$. Since these four terms sum to zero, we must have

$$\begin{aligned}
X_B &= \Theta(1), & \frac{1}{\mu} (X_N - X_U^T X_B^{-1} X_U) &= \Theta(1), \\
Z_N &= \Theta(1), & \frac{1}{\mu} (Z_B - Z_U Z_N^{-1} Z_U^T) &= \Theta(1).
\end{aligned}$$

Together with (3.1), this implies

$$X_N = \Theta(\mu), \quad Z_B = \Theta(\mu).$$

This completes the proof of the lemma. $\square$

Lemma 3.2 provides a precise result on the order of the eigenvalues of $X_B(\mu)$, $X_N(\mu)$, $Z_B(\mu)$, and $Z_N(\mu)$. We will now prove a preliminary result on the order of the off-diagonal blocks $X_U(\mu)$ and $Z_U(\mu)$.

LEMMA 3.3. *For $\mu \in (0, 1)$, there holds*

$$(3.2) \quad \begin{aligned} \|X_U(\mu)\| &= \Theta(1) \|Z_U(\mu)\|, \\ -X_U(\mu) \bullet Z_U(\mu) &= \Theta(1) \|X_U(\mu)\|^2, \\ \|X_U(\mu)\| &= o(\sqrt{\mu}), \quad \|Z_U(\mu)\| = o(\sqrt{\mu}) \quad \text{as } \mu \rightarrow 0. \end{aligned}$$

Proof. By the central path definition, we have

$$\mu I = \begin{bmatrix} X_B(\mu) & X_U(\mu) \\ X_U(\mu)^T & X_N(\mu) \end{bmatrix} \begin{bmatrix} Z_B(\mu) & Z_U(\mu) \\ Z_U(\mu)^T & Z_N(\mu) \end{bmatrix}.$$

Expanding the right-hand side and comparing the upper-right corner of the above identity, we have

$$(3.3) \quad 0 = X_B(\mu)Z_U(\mu) + X_U(\mu)Z_N(\mu),$$

or equivalently,

$$(3.4) \quad Z_U(\mu) = -X_B(\mu)^{-1}X_U(\mu)Z_N(\mu).$$

Using $X_B(\mu) = \Theta(1)$ and $Z_N(\mu) = \Theta(1)$ (see Lemma 3.2), this implies that

$$\|Z_U(\mu)\| = \Theta(1) \|X_U(\mu)\|.$$

This proves the first part of the lemma.

We now prove (3.2). Premultiplying both sides of (3.4) by the matrix $X_U(\mu)^T$ yields

$$X_U(\mu)^T Z_U(\mu) = -X_U(\mu)^T X_B(\mu)^{-1} X_U(\mu) Z_N(\mu).$$

Now taking the trace of the above matrices, we obtain

$$\begin{aligned} X_U(\mu) \bullet Z_U(\mu) &= -\operatorname{tr} X_U(\mu)^T X_B(\mu)^{-1} X_U(\mu) Z_N(\mu) \\ &= -\operatorname{tr} Z_N(\mu)^{1/2} X_U(\mu)^T X_B(\mu)^{-1} X_U(\mu) Z_N(\mu)^{1/2} \\ &= -\Theta(1) \|X_U(\mu)\|^2, \end{aligned}$$

where we used the fact that $X_B(\mu) = \Theta(1)$ and $Z_N(\mu) = \Theta(1)$, as shown in Lemma 3.2. This establishes (3.2).

It remains to prove the last part of the lemma. We consider an arbitrary convergent sequence $\{(X(\mu_k), Z(\mu_k)) : k = 1, 2, \dots\}$ on the central path with $\mu_k \rightarrow 0$; its limit is denoted by X^* , Z^* , so that

$$(3.5) \quad X(\mu_k) - X^* = o(1), \quad Z(\mu_k) - Z^* = o(1).$$

By Lemma 3.2, we have $X_N^* = 0$, $X_U^* = Z_U^* = 0$ and $Z_B^* = 0$. Since $(X(\mu_k) - X^*) \perp (Z(\mu_k) - Z^*)$, we have

$$(3.6) \quad \begin{aligned} 0 &= (X_B(\mu_k) - X_B^*) \bullet Z_B(\mu_k) + 2X_U(\mu_k) \bullet Z_U(\mu_k) \\ &\quad + X_N(\mu_k) \bullet (Z_N(\mu_k) - Z_N^*). \end{aligned}$$

Using (3.2) and (3.6), we obtain

$$(3.7) \quad \begin{aligned} \|X_U(\mu_k)\|^2 &= -\Theta(1) (X_U(\mu_k) \bullet Z_U(\mu_k)) \\ &= \Theta(1) ((X_B(\mu_k) - X_B^*) \bullet Z_B(\mu_k) + X_N(\mu_k) \bullet (Z_N(\mu_k) - Z_N^*)) \\ &= o(\mu_k), \end{aligned}$$

where in the last step we used (3.5) and of Lemma 3.2. This implies that $\|X_U(\mu)\|^2 = o(\mu)$ holds true on the entire central path curve, for otherwise there would exist a convergent subsequence $\{(X(\mu_k), Z(\mu_k)) : k = 1, 2, \dots\}$ for which

$$\liminf_{k \rightarrow \infty} \frac{\|X_U(\mu_k)\|^2}{\mu_k} > 0,$$

contradicting (3.7). The proof is complete. $\square$

We now use Lemma 3.2 and Lemma 3.3 to prove the convergence of the central path $\{(X(\mu), Z(\mu)) : \mu > 0\}$ to the analytic center (X^a, Z^a) , and to estimate the rate at which it converges to this limit.

LEMMA 3.4. *The primal-dual central path $\{(X(\mu), Z(\mu)) : \mu > 0\}$ converges to the analytic centers (X^a, Z^a) of $\mathcal{F}_P^*$ and $\mathcal{F}_D^*$, respectively. Moreover, if we let*

$$\epsilon(\mu) := \frac{\|X_U(\mu)\|}{\sqrt{\mu}},$$

then

$$\|X_B(\mu) - X_B^a\| = O((\epsilon(\mu) + \sqrt{\mu})^2), \quad \|Z_N(\mu) - Z_N^a\| = O((\epsilon(\mu) + \sqrt{\mu})^2).$$

Proof. Suppose $0 < \mu < 1$. By expanding $X(\mu)Z(\mu) = \mu I$ and comparing the upper-left block, we obtain

$$\mu I_B = X_B(\mu)Z_B(\mu) + X_U(\mu)Z_U(\mu)^T.$$

Premultiplying both sides with $(\mu X_B(\mu))^{-1}$ yields

$$(3.8) \quad X_B(\mu)^{-1} = \frac{1}{\mu} Z_B(\mu) + \frac{1}{\mu} X_B(\mu)^{-1} X_U(\mu) Z_U(\mu)^T.$$

Let $\mathcal{J}$ be an index set of minimal cardinality such that

$$\text{Span}\{A_B^{(i)} : i \in \mathcal{J}\} = \text{Span}\{A_B^{(i)} : i = 1, 2, \dots, m\}.$$

Since $Z_B^* = 0$, it follows from dual feasibility and (3.8) that

$$(3.9) \quad \begin{aligned} \frac{1}{\mu} Z_B(\mu) &= \sum_{i \in \mathcal{J}} \nu_i(\mu) A_B^{(i)} \quad \text{for some scalars } \nu_i(\mu) \\ &= X_B(\mu)^{-1} - \frac{1}{\mu} X_B(\mu)^{-1} X_U(\mu) Z_U(\mu)^T. \end{aligned}$$

From Lemma 3.2, we know that $Z_B(\mu)/\mu = \Theta(1)$. As the matrices $A_B^{(i)}$, $i \in \mathcal{J}$ are linearly independent, this implies that the sequences $\{\nu_i(\mu) : \mu \in (0, 1)\}$, $i \in \mathcal{J}$, are bounded. Moreover, we have from Lemma 3.3 that

$$\lim_{\mu \rightarrow 0} \frac{1}{\mu} X_B(\mu)^{-1} X_U(\mu) Z_U(\mu)^T = 0.$$

Hence, any limit X^* , ν_i^* , $i \in \mathcal{J}$ for $\mu \rightarrow 0$ satisfies the following nonlinear system of equations:

$$(3.10) \quad \begin{cases} X_B^{-1} - \sum_{i \in \mathcal{J}} \nu_i A_B^{(i)} = 0, \\ A_B^{(i)} \bullet X_B = b_i, \quad i \in \mathcal{J}. \end{cases}$$

Moreover, since $Z_B(\mu)/\mu = \Theta(1)$ and $X_B(\mu) = \Theta(1)$ for $\mu \in (0, 1)$, we have

$$\sum_{i \in \mathcal{J}} \nu_i^* A_B^{(i)} \succ 0, \quad X_B^* \succ 0.$$

By (2.3), this means that $X^* = X^a$, the analytic center of $\mathcal{F}_P^*$ and hence

$$X(\mu) - X^a = o(1), \quad \text{as } \mu \rightarrow 0.$$

Using the linear independence of the matrices $A_B^{(i)}$, $i \in \mathcal{J}$ and using the fact that X_B^a is positive definite, it can be checked that the Jacobian (with respect to the variables X_B and ν_i , $i \in \mathcal{J}$) of the nonlinear system (3.10) is nonsingular at the solution X_B^a , ν_i^* , $i \in \mathcal{J}$. Hence we can apply the classical inverse function theorem to the above nonlinear system at the point: $X_B = X_B^a$, $\nu_i = \nu_i^*$, $i \in \mathcal{J}$, to obtain

(3.11)

$$\|X_B(\mu) - X_B^a\| = O\left(\|X_B(\mu)^{-1} - \sum_{i \in \mathcal{J}} \nu_i(\mu) A_B^{(i)}\| + \sum_{i \in \mathcal{J}} |A_B^{(i)} \bullet X_B(\mu) - b_i|\right).$$

By (3.9) and the definition of $\epsilon(\mu)$, we obtain from Lemma 3.3

$$\left\| X_B(\mu)^{-1} - \sum_{i \in \mathcal{J}} \nu_i(\mu) A_B^{(i)} \right\| = \left\| \frac{1}{\mu} X_B(\mu)^{-1} X_U(\mu) Z_U(\mu)^T \right\| = O(\epsilon(\mu)^2).$$

Also we have from $X(\mu) \in \mathcal{F}_P$

$$\begin{aligned} |A_B^{(i)} \bullet X_B(\mu) - b_i| &= |2A_U^{(i)} \bullet X_U(\mu) + A_N^{(i)} \bullet X_N(\mu)| \\ &= O(\epsilon(\mu)\sqrt{\mu} + \mu) \quad \text{for } i \in \mathcal{J}. \end{aligned}$$

Substituting the above two bounds into (3.11) yields

$$\|X_B(\mu) - X_B^a\| = O((\epsilon(\mu) + \sqrt{\mu})^2).$$

It can be shown by an analogous argument that

$$\|Z_N(\mu) - Z_N^a\| = O((\epsilon(\mu) + \sqrt{\mu})^2).$$

The proof is complete. $\square$

Lemma 3.4 only provides a rough sketch of the convergence behavior of the central path as $\mu \rightarrow 0$. Our goal is to characterize this convergence behavior more precisely.

THEOREM 3.5. *Let $\mu \in (0, 1)$. There holds*

$$(3.12) \quad X_B(\mu) = \Theta(1), \quad Z_N(\mu) = \Theta(1), \quad X_N(\mu) = \Theta(\mu), \quad Z_B(\mu) = \Theta(\mu),$$

and

$$(3.13) \quad \|X(\mu) - X^a\| = O(\mu), \quad \|Z(\mu) - Z^a\| = O(\mu).$$

Proof. The estimate (3.12) is already known from Lemma 3.2, so we only need to prove (3.13). By Lemma 3.3 and Lemma 3.4, it is sufficient to show that

$$\|X_U(\mu)\| = O(\mu).$$

Suppose to the contrary that there exists a sequence

$$(3.14) \quad \{(X(\mu_k), Z(\mu_k)) : k = 1, 2, \dots\}$$

with $\|X_U(\mu_k)\| > 0$ for all k and

$$(3.15) \quad \lim_{k \rightarrow \infty} \frac{\mu_k}{\|X_U(\mu_k)\|} = 0.$$

With the notation of Lemma 3.4, the condition (3.15) implies

$$(3.16) \quad \epsilon(\mu_k) + \sqrt{\mu_k} \sim \epsilon(\mu_k) = \frac{\|X_U(\mu_k)\|}{\sqrt{\mu_k}}.$$

By virtue of Lemma 3.2, we can choose the subsequence (3.14) such that

$$\lim_{k \rightarrow \infty} \frac{Z_B(\mu_k)}{\mu_k}$$

exists, and using Lemma 3.4 and relation (3.16), we can also assume the existence of

$$\Delta_B^x(\infty) := \lim_{k \rightarrow \infty} \frac{\mu_k}{\|X_U(\mu_k)\|^2} (X_B(\mu_k) - X_B^a).$$

From the existence of the above limits, we obtain

$$(3.17) \quad \lim_{k \rightarrow \infty} \frac{(X_B(\mu_k) - X_B^a) \bullet Z_B(\mu_k)}{\|X_U(\mu_k)\|^2} = \lim_{k \rightarrow \infty} \frac{\Delta_B^x(\infty) \bullet Z_B(\mu_k)}{\mu_k}.$$

Notice that the hypothesis (3.15) implies that

$$\lim_{k \rightarrow \infty} \frac{\mu_k}{\|X_U(\mu_k)\|^2} (X(\mu_k) - X^a) = \begin{bmatrix} \Delta_B^x(\infty) & 0 \\ 0 & 0 \end{bmatrix}.$$

Using also $Z_B^a = 0$, we thus obtain for any $k = 1, 2, \dots$ that

$$\begin{aligned} \frac{\Delta_B^x(\infty) \bullet Z_B(\mu_k)}{\mu_k} &= \frac{\Delta_B^x(\infty) \bullet (Z_B(\mu_k) - Z_B^a)}{\mu_k} \\ &= \lim_{j \rightarrow \infty} \frac{\mu_j}{\mu_k \|X_U(\mu_j)\|^2} ((X(\mu_j) - X^a) \bullet (Z(\mu_k) - Z^a)) \\ &= 0, \end{aligned}$$

where the last step is due to the orthogonality condition $(X(\mu_j) - X^a) \perp (Z(\mu_k) - Z^a)$ for all j and k . Therefore,

$$(3.18) \quad 0 = \lim_{k \rightarrow \infty} \frac{\Delta_B^x(\infty) \bullet Z_B(\mu_k)}{\mu_k} = \lim_{k \rightarrow \infty} \frac{(X_B(\mu_k) - X_B^a) \bullet Z_B(\mu_k)}{\|X_U(\mu_k)\|^2},$$

where we used (3.17). Analogously, it can be shown that

$$(3.19) \quad \lim_{k \rightarrow \infty} \frac{X_N(\mu_k) \bullet (Z_N(\mu_k) - Z_N^a)}{\|X_U(\mu_k)\|^2} = 0.$$

Since $(X(\mu_k) - X^a) \perp (Z(\mu_k) - Z^a)$, we have from (3.18) and (3.19) that

$$\begin{aligned} 0 &= \lim_{k \rightarrow \infty} \frac{(X(\mu_k) - X^a) \bullet (Z(\mu_k) - Z^a)}{\|X_U(\mu_k)\|^2} \\ &= \lim_{k \rightarrow \infty} 2 \frac{X_U(\mu_k) \bullet Z_U(\mu_k)}{\|X_U(\mu_k)\|^2}, \end{aligned}$$

which clearly contradicts (3.2). The proof is complete. $\square$

Theorem 3.5 characterizes completely the limiting behavior of the primal-dual central path as $\mu \rightarrow 0$. We point out that this limiting behavior was well understood in the context of linear programming and the monotone horizontal linear complementarity problem, see Güler [3] and Monteiro and Tsuchiya [13], respectively. Notice that under a *nondegeneracy assumption* (the reader is referred to Alizadeh, Haeberly, and Overton [2] or Kojima, Shida, and Shindoh [7] for a discussion on this concept), i.e., the Jacobian of the nonlinear system

$$\begin{cases} A^{(i)} \bullet X = b_i, & i = 1, 2, \dots, m, \\ \sum_{i=1}^m y_i A^{(i)} + Z = C, \\ XZ + ZX = 0, \\ X \in \mathcal{S}, Z \in \mathcal{S} \end{cases}$$

is nonsingular at (X^a, y^a, Z^a) , the estimates (3.13) follow immediately from the application of the classical inverse function theorem. Thus, the real contribution of Theorem 3.5 lies in establishing these estimates in the absence of the nondegeneracy assumption.

It is known that in the case of linear programming the proof of quadratic convergence of predictor-corrector interior point algorithms required an error bound result of Hoffman. This error bound states that the distance from any vector $x \in \mathbb{R}^n$ to a polyhedral set $\mathcal{P} := \{x : Ax \leq a\}$ can be bounded in terms of the “amount of constraint violation” at x , namely $\|[Ax - a]_+\|$, where $[\cdot]_+$ denotes the positive part of a vector. More precisely, Hoffman’s error bound ([5]) states that there exists some constant $\tau > 0$ such that

$$\text{dist}(x, \mathcal{P}) \leq \tau \|[Ax - a]_+\| \quad \forall x \in \mathbb{R}^n.$$

Unfortunately, this error bound no longer holds for linear systems over the cone of positive semidefinite matrices (see the example below). In fact, much of the difficulty in the local analysis of interior point algorithms for SDP can be attributed to this lack

of an analogue of Hoffman's error bound result (see the analysis of [7, 16]). Specifically, without such an error bound result, it is difficult to estimate the distance from the current iterates to the optimal solution set. In essence, what we have established in Theorem 3.5 is an error bound result along the central path. In other words, although a Hoffman-type error bound cannot hold over the entire feasible set of (P), it nevertheless still holds true on the restricted region "near the central path." One consequence of this restriction to the central path is that we will need to require the iterates to stay "sufficiently close" to the central path in order to establish the super-linear convergence of the algorithm. Such a requirement on the iterates was called "tangential convergence to the optimal solution set" by Kojima, Shida, and Shindoh [7]. Notice that the analysis in this reference required the additional nondegeneracy assumption to establish their superlinear convergence result. In contrast, this assumption is no longer needed in our analysis because Theorem 3.5 holds without the nondegeneracy assumption.

Example. Consider the following linear system over the cone of positive semidefinite matrices in $\mathbb{R}^{2 \times 2}$:

$$X_{11} = 0, \quad X_{22} = 1, \quad X = \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \succeq 0.$$

Clearly, there is exactly one solution X^* to the above linear system, namely,

$$X^* := \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

For each $\epsilon > 0$, consider the matrix

$$X(\epsilon) := \begin{bmatrix} \epsilon^2 & \epsilon \\ \epsilon & 1 \end{bmatrix}.$$

Clearly, $X(\epsilon) \succeq 0$. The amount of constraint violation is equal to ϵ^2 . However, the distance $\|X(\epsilon) - X^*\|_F = \Theta(\epsilon)$. Thus, there cannot exist some fixed $\tau > 0$ such that $\|X(\epsilon) - X^*\| \leq \tau \epsilon^2$, for all $\epsilon > 0$.

4. A polynomial predictor-corrector algorithm. In [14], Nesterov and Todd developed a theoretical foundation for interior point methods for self-scaled cones. This framework was then used in [15] to generalize primal-dual interior point methods from linear programming to conic convex programming over self-scaled cones. The results of Nesterov and Todd can be specialized to SDP, since the cone $\mathcal{S}_+$ of positive semidefinite matrices is self-scaled. In fact, the main results in Sturm and Zhang [17] can be viewed as such a specialization. However, unlike Nesterov and Todd [14, 15], Sturm and Zhang [17] adopt the V -space framework, thus generalizing the approach of Kojima et al. [6]. We begin by summarizing some of the results on V -space path following for SDP that were obtained in [17].

Let $(X, Z) \in \mathcal{F}_P \times \mathcal{F}_D$ with $X \succ 0$, $Z \succ 0$. Then, there exists a unique positive definite matrix $D \in \mathcal{S}_{++}$ such that [17, Lemma 2.1]

$$(4.1) \quad X = DZD.$$

Let L be such that

$$(4.2) \quad LL^T = D,$$

and let $V := L^T Z L$. It follows that

$$V = L^{-1} X L^{-T} = L^T Z L.$$

The quantity

$$\delta(X, Z) := \left\| I - \frac{1}{\mu} L^{-1} X Z L \right\|_F$$

serves as a centrality measure, with $\mu := X \bullet Z/n$. Considering the identity

$$\delta(X, Z) = \left\| I - \frac{1}{\mu} L^{-1} X Z L \right\|_F = \left\| I - \frac{1}{\mu} X^{1/2} Z X^{1/2} \right\|_F,$$

it becomes apparent that $\delta(X, Z)$ is the same centrality measure used in many other papers, including [8] and [12]. It is easy to see that the central path is the set of solutions (X, Z) with $\delta(X, Z) = 0$ or, equivalently, those solutions for which $V = \sqrt{\mu} I$. Moreover, we have

$$(4.3) \quad (1 + \delta(X, Z))I \succeq \frac{1}{\mu} V^2 \succeq (1 - \delta(X, Z))I.$$

In V -space path following, we want to drive the V -iterates towards the origin by Newton's method, in such a way that the iterates reside in a cone around the identity matrix. Before stating the Newton equation, we need to introduce the linear space $\mathcal{A}(L)$,

$$\mathcal{A}(L) := \text{Span}\{L^T A^{(i)} L : i = 1, 2, \dots, m\}$$

and its orthogonal complement in $\mathcal{S}$

$$\mathcal{A}^\perp(L) := \{\Delta X \in \mathcal{S} : (L^T A^{(i)} L) \bullet \Delta X = 0 \text{ for } i = 1, 2, \dots, m\}.$$

A Newton direction for obtaining a $(\gamma\mu)$ -center, for some $\gamma \in [0, 1]$, is the solution $(\Delta X, \Delta Z)$ of the following system of linear equations [17, equation (17)]:

$$(4.4) \quad \begin{cases} \Delta X + D \Delta Z D = \gamma \mu Z^{-1} - X, \\ \Delta X \in \mathcal{A}^\perp(L), \quad \Delta Z \in \mathcal{A}(L). \end{cases}$$

For $\gamma = 0$, we denote the solution of (4.4) by $(\Delta X^p, \Delta Z^p)$, the predictor direction. For $\gamma = 1$, the solution is denoted by $(\Delta X^c, \Delta Z^c)$, the corrector direction. If we let

$$\Delta \bar{X} := L^{-1} \Delta X L^{-T}, \quad \Delta \bar{Z} := L^T \Delta Z L,$$

then we can rewrite (4.4) as

$$\begin{cases} \Delta \bar{X} + \Delta \bar{Z} = \gamma \mu V^{-1} - V, \\ \Delta \bar{X} \in \mathcal{A}^\perp(L), \quad \Delta \bar{Z} \in \mathcal{A}(L). \end{cases}$$

It follows from orthogonality that

$$(4.5) \quad \|\Delta \bar{X}^p\|_F^2 + \|\Delta \bar{Z}^p\|_F^2 = \|V\|_F^2 = n\mu.$$

The corrector direction does not change the duality gap

$$(4.6) \quad (X + \Delta X^c) \bullet (Z + \Delta Z^c) = X \bullet Z,$$

whereas

$$(4.7) \quad (X + t\Delta X^p) \bullet (Z + t\Delta Z^p) = (1 - t)X \bullet Z$$

for any $t \in \mathbb{R}$; see equation (16) of [17].

Algorithm SDP(ϵ).

Given $(X^0, Z^0) \in \mathcal{F}_P \times \mathcal{F}_D$ with $\delta(X^0, Z^0) \leq 1/4$.

Parameter ϵ , $0 < \epsilon \leq (X^0 \bullet Z^0)/n$ and positive integer r .

Let $k = 0$.

REPEAT (main iteration)

Let $X = X^k, Z = Z^k$ and $\mu_k = X \bullet Z/n$.

Predictor: compute $(\Delta X^p, \Delta Z^p)$ from (4.4) with $\gamma = 0$.

Compute the largest step t_k such that for all $0 \leq t \leq t_k$ there holds

$$\delta(X + t\Delta X^p, Z + t\Delta Z^p) \leq \min(1/2, ((1 - t)\mu_k/\epsilon)^{2^{-r}}).$$

Let $X' := X + t_k\Delta X^p, Z' := Z + t_k\Delta Z^p$ and $\beta_k = \min(1/4, (1 - t_k)\mu_k/\epsilon)$.

Corrector:

FOR $i = 1$ **to** r **DO**

Let $X = X', Z = Z'$.

IF $\delta(X, Z) \leq \beta_k$ **THEN** exit loop.

Compute $(\Delta X^c, \Delta Z^c)$ from (4.4) with $\gamma = 1$.

Set $X' = X + \Delta X^c, Z' = Z + \Delta Z^c$.

END FOR

$X^{k+1} = X', Z^{k+1} = Z'$

Set $k = k + 1$.

UNTIL convergence.

Interestingly, each corrector step reduces $\delta(\cdot, \cdot)$ at a quadratic rate as stated in the following result.

LEMMA 4.1. *If $\delta(X, Z) \leq 1/2$, then*

$$\delta(X + \Delta X^c, Z + \Delta Z^c) \leq \delta(X, Z)^2.$$

Proof. It follows from Lemma 4.5 in [17] that

$$X + \Delta X^c \succ 0, \quad Z + \Delta Z^c \succ 0.$$

Hence, the desired result is an immediate consequence of Lemma 4.4 in [17]. $\square$

Lemma 4.1 implies that for any $k \geq 1$, we have

$$(4.8) \quad \delta(X^k, Z^k) \leq \beta_{k-1}.$$

Also, it follows from (4.6) and (4.7) that for any $k > 1$

$$(4.9) \quad \begin{aligned} \delta(X^k, Z^k) &\leq \beta_{k-1} \leq (1 - t_{k-1})\mu_{k-1}/\epsilon \\ &= \mu_k/\epsilon = O(\mu_k). \end{aligned}$$

Furthermore, if $\beta_k = 1/4$, then only one (instead of r) corrector step is needed to recenter the iterate (see [17]). In other words, the iterations of Algorithm SDP(ϵ) are

identical to those of the primal-dual predictor-corrector algorithm of [17], for all k with

$$\frac{\mu_k}{\epsilon} \geq \frac{1}{4}.$$

We can therefore conclude from Theorem 5.2 in [17] that the algorithm yields $\mu_k \leq \epsilon/4$ for all $k \geq \Gamma\sqrt{n}\log(\mu_0/\epsilon)$, where Γ is a universal constant, independent of the problem data. Thus, we have the following polynomial complexity result.

THEOREM 4.2. *For each $0 < \epsilon < (X_0 \bullet Z_0)/n$, Algorithm $SDP(\epsilon)$ will generate an iterate $(X^k, Z^k) \in \mathcal{F}_P \times \mathcal{F}_D$ with $(X^k \bullet Z^k)/n \leq \epsilon/4$ in at most $\Gamma\sqrt{n}\log(\mu_0/\epsilon)$ predictor-corrector steps, where Γ is a constant that is independent of the problem data.*

In addition to having polynomial complexity, Algorithm $SDP(\epsilon)$ also possesses a superlinear rate of convergence. We prove this in the next section.

5. Convergence analysis. We begin by establishing the global convergence of Algorithm $SDP(\epsilon)$. Notice that Algorithm $SDP(\epsilon)$ chooses the predictor step length t_k to be the largest step such that for all $0 \leq t \leq t_k$ there holds

$$(5.1) \quad \delta(X + t\Delta X^p, Z + t\Delta Z^p) \leq \min\left(\frac{1}{2}, ((1-t)\mu/\epsilon)^{2^{-r}}\right).$$

It was shown in [17] (equation (21) therein) that

$$(5.2) \quad (1-t)\delta(X + t\Delta X^p, Z + t\Delta Z^p) \leq (1-t)\delta(X, Z) + t^2 \|\Delta \bar{X}^p \Delta \bar{Z}^p\|_F / \mu.$$

Using (4.5), we thus obtain for $0 \leq t < 1$ that

$$(5.3) \quad \delta(X + t\Delta X^p, Z + t\Delta Z^p) \leq \delta(X, Z) + \frac{nt^2}{2(1-t)}.$$

Combining (5.1) and (5.3), we can easily establish the global convergence of Algorithm $SDP(\epsilon)$.

THEOREM 5.1. *There holds*

$$\lim_{k \rightarrow \infty} \mu_k = 0,$$

i.e., Algorithm $SDP(\epsilon)$ is globally convergent.

Proof. Due to (4.7), the sequence $\mu_0, \mu_1, \dots$ is a monotonically decreasing sequence. Hence, $\lim_{k \rightarrow \infty} \mu_k$ exists. Suppose contrary to the statement of the lemma that

$$(5.4) \quad \mu_\infty = \lim_{k \rightarrow \infty} \mu_k, \quad \mu_\infty > 0.$$

Consider $k \geq \Gamma\sqrt{n}\log(\mu_0/\epsilon)$. From Theorem 4.2, we have

$$(5.5) \quad \mu_k \leq \epsilon/4$$

and hence, using (4.8),

$$(5.6) \quad \delta(X^k, Z^k) \leq \beta_{k-1} = \min\left(\frac{1}{4}, \frac{\mu_k}{\epsilon}\right) = \frac{\mu_k}{\epsilon}.$$

Now consider a step length $0 \leq t \leq 0.5\sqrt{\mu_k/(n\epsilon)}$ and note from (5.5) that

$$1 - t \geq \frac{3}{4}.$$

We obtain from (5.3) and (5.6) that

$$\begin{aligned} \delta(X^k + t(\Delta X^p)^k, Z^k + t(\Delta Z^p)^k) &\leq \delta(X^k, Z^k) + \frac{nt^2}{2(1-t)} \\ &\leq \frac{\mu_k}{\epsilon} + \frac{\mu_k}{6\epsilon} \\ &\leq \frac{7}{6} \cdot \frac{4}{3} \cdot \frac{(1-t)\mu_k}{\epsilon} \\ &< \left(\frac{(1-t)\mu_k}{\epsilon} \right)^{2^{-r}}, \end{aligned}$$

where we used (5.5) and the fact that $r \geq 1$. By definition of t_k , this implies that

$$t_k \geq \frac{1}{2} \sqrt{\frac{\mu_k}{n\epsilon}} \geq \frac{1}{2} \sqrt{\frac{\mu_\infty}{n\epsilon}} = \Theta(1).$$

This, together with (4.7), implies that $1 - \mu_{k+1}/\mu_k = \Theta(1)$, which contradicts (5.4). $\square$

Next we proceed to establish the superlinear convergence of Algorithm SDP(ϵ). In light of (4.7), we need only to show that the predictor step length t_k approaches 1. Hence we are led to bound t_k from below. For this purpose, we note from (5.2) that, for $t \in (0, 1)$,

$$(5.7) \quad \delta(X + t\Delta X^p, Z + t\Delta Z^p) \leq \delta(X, Z) + \frac{1}{1-t} \|\Delta \bar{X}^p \Delta \bar{Z}^p\|_F / \mu.$$

Thus, if we can properly bound $\|\Delta \bar{X}^p \Delta \bar{Z}^p\|_F$, then we will obtain a lower bound on the predictor step length t_k .

To begin, let us consider L_μ with

$$L_\mu L_\mu^T = \frac{1}{\sqrt{\mu}} X(\mu).$$

Remark that

$$\sqrt{\mu} I = L_\mu^{-1} X(\mu) L_\mu^{-T} = L_\mu^T Z(\mu) L_\mu.$$

Now define the predictor direction starting from the solution $(X(\mu), Z(\mu))$ on the central path as follows:

$$\begin{cases} \Delta \hat{X}^p(\mu) + \Delta \hat{Z}^p(\mu) = -\sqrt{\mu} I, \\ \Delta \hat{X}^p(\mu) \in \mathcal{A}^\perp(L_\mu), \quad \Delta \hat{Z}^p(\mu) \in \mathcal{A}(L_\mu). \end{cases}$$

Let $(\hat{X}^a, \hat{Z}^a)$ be the analytic center of the optimal solution set in the L_μ -transformed space,

$$\hat{X}^a := L_\mu^{-1} X^a L_\mu^{-T}, \quad \hat{Z}^a := L_\mu^T Z^a L_\mu.$$

We will show in Lemma 5.2 below that $\Delta\hat{X}^p(\mu)$ is close to the optimal step $\hat{X}^a - \sqrt{\mu}I$ for small μ . We will bound the difference between $\Delta\hat{X}^p(\mu)$ and $\Delta\bar{X}^p$ afterwards.

LEMMA 5.2. *There holds*

$$\left\| \sqrt{\mu}I + \Delta\hat{X}^p(\mu) - \hat{X}^a \right\| + \left\| \sqrt{\mu}I + \Delta\hat{Z}^p(\mu) - \hat{Z}^a \right\| = O(\mu^{3/2}).$$

Proof. Since

$$\hat{X}^a \hat{Z}^a = L_\mu^{-1} X^a Z^a L_\mu = 0,$$

it follows that

$$(\sqrt{\mu}I - \hat{X}^a)(\sqrt{\mu}I - \hat{Z}^a) = (\sqrt{\mu}I - \hat{Z}^a)(\sqrt{\mu}I - \hat{X}^a).$$

Therefore, the matrix $(\sqrt{\mu}I - \hat{X}^a)(\sqrt{\mu}I - \hat{Z}^a)$, or equivalently, the matrix

$$L_\mu^{-1}(X(\mu) - X^a)(Z(\mu) - Z^a)L_\mu,$$

is symmetric. By the property of the F -norm, we obtain

$$\begin{aligned} \left\| (\sqrt{\mu}I - \hat{X}^a)(\sqrt{\mu}I - \hat{Z}^a) \right\|_F &= \left\| L_\mu^{-1}(X(\mu) - X^a)(Z(\mu) - Z^a)L_\mu \right\|_F \\ &= (\text{tr}(X(\mu) - X^a)(Z(\mu) - Z^a))^{1/2} \\ &= O(\mu^2), \end{aligned} \tag{5.8}$$

where the last step follows from Theorem 3.5. Now since $\hat{X}^a \hat{Z}^a = 0$ and $\Delta\hat{X}^p(\mu) + \Delta\hat{Z}^p(\mu) = -\sqrt{\mu}I$, we have

$$\begin{aligned} (\sqrt{\mu}I - \hat{X}^a)(\sqrt{\mu}I - \hat{Z}^a) &= \mu I - \sqrt{\mu}(\hat{X}^a + \hat{Z}^a) \\ &= \sqrt{\mu}(\sqrt{\mu}I + \Delta\hat{X}^p(\mu) - \hat{X}^a) \\ &\quad + \sqrt{\mu}(\sqrt{\mu}I + \Delta\hat{Z}^p(\mu) - \hat{Z}^a). \end{aligned}$$

As

$$\sqrt{\mu}I + \Delta\hat{X}^p(\mu) - \hat{X}^a \in \mathcal{A}^\perp(L_\mu), \quad \sqrt{\mu}I + \Delta\hat{Z}^p(\mu) - \hat{Z}^a \in \mathcal{A}(L_\mu),$$

it follows that

$$\begin{aligned} &\left\| \sqrt{\mu}I + \Delta\hat{X}^p(\mu) - \hat{X}^a \right\|_F^2 + \left\| \sqrt{\mu}I + \Delta\hat{Z}^p(\mu) - \hat{Z}^a \right\|_F^2 \\ &= \frac{1}{\mu} \left\| (\hat{X}^a - \sqrt{\mu}I)(\hat{Z}^a - \sqrt{\mu}I) \right\|_F^2 = O(\mu^3), \end{aligned}$$

where the last step is due to (5.8). This proves the lemma. $\square$

Lemma 5.2 applies only to $(\Delta\hat{X}^p(\mu), \Delta\hat{Z}^p(\mu))$, namely the predictor directions for the points located exactly on the central path. What we need is a similar bound for $(\Delta\bar{X}^p, \Delta\bar{Z}^p)$ (obtained at points close to the central path). This leads us to bound the difference $\Delta\hat{X}^p(\mu) - \Delta\bar{X}^p$. Indeed, our next goal is to show (Lemma 5.6) that

$$\left\| \Delta\hat{X}^p(\mu) - \Delta\bar{X}^p \right\|_F = O(\sqrt{\mu}\delta(X, Z)).$$

We prove this bound by a sequence of lemmas. Let D be given by (4.1) and define

$$\bar{D} := L_\mu^{-1} D L_\mu^{-T},$$

so that $\bar{D} = I$ if $X = X(\mu)$ and $Z = Z(\mu)$. Choose L by

$$L := L_\mu \bar{D}^{1/2},$$

and notice that indeed $LL^T = D$, as stipulated by (4.2).

LEMMA 5.3. *Suppose $\delta(X, Z) \leq 1/2$. There holds*

$$\|L^{-1}(X(\mu) - X)L^{-T}\| + \|L^T(Z(\mu) - Z)L\| = O(\sqrt{\mu}\delta(X, Z)).$$

Proof. Let

$$\Delta_x(\mu) := L^{-1}(X(\mu) - X)L^{-T}, \quad \Delta_z(\mu) := L^T(Z(\mu) - Z)L.$$

Clearly, $\Delta_x(\mu)$ and $\Delta_z(\mu)$ are symmetric and $\Delta_x(\mu) \perp \Delta_z(\mu)$. Let ρ denote the smallest eigenvalue of $\Delta_x(\mu) + \Delta_z(\mu)$, i.e.,

$$\rho = \arg \max\{\bar{\rho} : \Delta_x(\mu) + \Delta_z(\mu) \succeq \bar{\rho}I\}.$$

Since $X \bullet Z = X(\mu) \bullet Z(\mu) = n\mu$, we have

$$\begin{aligned} \text{tr}(Z(X(\mu) - X) + X(Z(\mu) - Z)) &= \text{tr}((X(\mu) - X)Z + X(Z(\mu) - Z)) \\ &= -\text{tr}((X(\mu) - X)(Z(\mu) - Z)) - \text{tr} XZ \\ &\quad + \text{tr} X(\mu)Z(\mu) \\ &= 0, \end{aligned}$$

where the last step follows from $(X(\mu) - X) \perp (Z(\mu) - Z)$. Recall that $V = L^{-1}XL^{-T} = L^TZL$. Consider

$$\begin{aligned} \text{tr}(V(\Delta_x(\mu) + \Delta_z(\mu))) &= \text{tr}(L^TZ(X(\mu) - X)L^{-T} + L^{-1}X(Z(\mu) - Z)L) \\ &= \text{tr}(Z(X(\mu) - X) + X(Z(\mu) - Z)) \\ &= 0. \end{aligned}$$

By (4.3), the matrix V is symmetric positive definite and $V = \Theta(\sqrt{\mu})$. Diagonalize the symmetric matrix $\Delta_x(\mu) + \Delta_z(\mu) = Q^T\Lambda Q$ and consider

$$0 = \text{tr}(V(\Delta_x(\mu) + \Delta_z(\mu))) = \text{tr}(VQ^T\Lambda Q) = \text{tr}(QVQ^T\Lambda).$$

Since $QVQ^T = \Theta(\sqrt{\mu})$, the diagonal entries of QVQ^T must be $\Theta(\sqrt{\mu})$. Therefore, the preceding equation implies that the diagonal matrix Λ must have a nonpositive eigenvalue and that its diagonal entries are of the same order of magnitude. In other words, $\rho \leq 0$ and $\|\Lambda\| = O(|\rho|)$. This further implies

$$(5.9) \quad \|\Delta_x(\mu) + \Delta_z(\mu)\| = O(|\rho|).$$

By the definition of the central path, we have

$$\begin{aligned} \mu I &= (V + \Delta_x(\mu))(V + \Delta_z(\mu)) \\ &= \left(V + \frac{\Delta_x(\mu) + \Delta_z(\mu)}{2} + \frac{\Delta_x(\mu) - \Delta_z(\mu)}{2}\right) \\ &\quad \times \left(V + \frac{\Delta_x(\mu) + \Delta_z(\mu)}{2} - \frac{\Delta_x(\mu) - \Delta_z(\mu)}{2}\right). \end{aligned}$$

Since the left-hand side matrix is symmetric, the skew-symmetric cross term must cancel when we expand the matrix product in the right-hand side. It follows that

$$\mu I = \left(V + \frac{\Delta_x(\mu) + \Delta_z(\mu)}{2} \right)^2 - \frac{1}{4} (\Delta_x(\mu) - \Delta_z(\mu))^2$$

and therefore,

$$V + \frac{\Delta_x(\mu) + \Delta_z(\mu)}{2} \succeq \sqrt{\mu} I.$$

Using (4.3), we obtain

$$|\rho| = O(\sqrt{\mu} \delta(X, Z)).$$

Combining this with (5.9) and using the fact that $\Delta_x(\mu) \perp \Delta_z(\mu)$, we have

$$\|\Delta_x(\mu)\| + \|\Delta_z(\mu)\| = O(|\rho|) = O(\sqrt{\mu} \delta(X, Z)). \quad \square$$

LEMMA 5.4. *Suppose $\delta(X, Z) \leq 1/2$. Then there holds*

$$\|\bar{D} - I\| = O(\delta(X, Z)).$$

Proof. Notice that

$$(5.10) \quad L_\mu^{-1} X L_\mu^{-T} = \sqrt{\mu} I + L_\mu^{-1} (X - X(\mu)) L_\mu^{-T}$$

and

$$(5.11) \quad L_\mu^T Z L_\mu = \sqrt{\mu} I + L_\mu^T (Z - Z(\mu)) L_\mu.$$

Now using

$$L_\mu^{-1} X L_\mu^{-T} = \bar{D} (L_\mu^T Z L_\mu) \bar{D},$$

we have, by premultiplying and postmultiplying (5.10) by $\bar{D}^{-1/2}$ and (5.11) by $\bar{D}^{1/2}$ and rearranging terms,

$$\sqrt{\mu} (\bar{D}^{-1} - \bar{D}) = L^{-1} (X(\mu) - X) L^{-T} + L^T (Z - Z(\mu)) L.$$

Together with Lemma 5.3, this implies $\bar{D} = \Theta(1)$ and

$$\|\bar{D} - I\| = O(\delta(X, Z)).$$

The lemma is proved. $\square$

Now, let

$$\Delta \hat{X}^p := \bar{D}^{1/2} \Delta \bar{X}^p \bar{D}^{1/2}, \quad \Delta \hat{Z}^p := \bar{D}^{-1/2} \Delta \bar{Z}^p \bar{D}^{-1/2}.$$

Notice that $(\Delta \hat{X}^p, \Delta \hat{Z}^p) \in \mathcal{A}^\perp(L_\mu) \times \mathcal{A}(L_\mu)$.

LEMMA 5.5. *Suppose $\delta(X, Z) \leq 1/2$. We have*

$$\left\| \Delta \hat{X}^p - \Delta \bar{X}^p \right\| + \left\| \Delta \hat{Z}^p - \Delta \bar{Z}^p \right\| = O(\sqrt{\mu} \delta(X, Z)).$$

Proof. We have

$$\Delta \hat{X}^p = \bar{D}^{1/2} \Delta \bar{X}^p \bar{D}^{1/2} = \Delta \bar{X}^p + (\bar{D}^{1/2} - I) \Delta \bar{X}^p \bar{D}^{1/2} + \Delta \bar{X}^p (\bar{D}^{1/2} - I).$$

Now using Lemma 5.4 and (4.5), we see that

$$\left\| \Delta \hat{X}^p - \Delta \bar{X}^p \right\| = O(\sqrt{\mu} \delta(X, Z)).$$

It can be shown in an analogous way that

$$\left\| \Delta \hat{Z}^p - \Delta \bar{Z}^p \right\| = O(\sqrt{\mu} \delta(X, Z)). \quad \square$$

Now we are ready to bound the difference between $\Delta \hat{X}^p(\mu)$ and $\Delta \bar{X}^p$.

LEMMA 5.6. *Suppose $\delta(X, Z) \leq 1/2$. We have*

$$\left\| \Delta \hat{X}^p(\mu) - \Delta \bar{X}^p \right\| + \left\| \Delta \hat{Z}^p(\mu) - \Delta \bar{Z}^p \right\| = O(\sqrt{\mu} \delta(X, Z)).$$

Proof. By definition of the predictor directions, we have

$$\Delta \hat{X}^p(\mu) + \Delta \hat{Z}^p(\mu) = -\sqrt{\mu} I$$

and

$$\Delta \bar{X}^p + \Delta \bar{Z}^p = -V.$$

Combining these two relations yields

$$\Delta \hat{X}^p(\mu) - \Delta \bar{X}^p + \Delta \hat{Z}^p(\mu) - \Delta \bar{Z}^p = V - \sqrt{\mu} I + \Delta \bar{X}^p - \Delta \hat{X}^p + \Delta \bar{Z}^p - \Delta \hat{Z}^p.$$

Now using Lemma 5.5 and using the fact that

$$\|V - \sqrt{\mu} I\| = \|(V + \sqrt{\mu} I)^{-1}(V^2 - \mu I)\| \leq \sqrt{\mu} \delta(X, Z),$$

we obtain

$$\left\| \Delta \hat{X}^p(\mu) - \Delta \bar{X}^p + \Delta \hat{Z}^p(\mu) - \Delta \bar{Z}^p \right\| = O(\sqrt{\mu} \delta(X, Z)).$$

Since $(\Delta \hat{X}^p(\mu) - \Delta \bar{X}^p) \perp (\Delta \hat{Z}^p(\mu) - \Delta \bar{Z}^p)$, the lemma follows from the above relation, after applying Lemma 5.5 once more. $\square$

Combining (5.8), Lemma 5.2, and Lemma 5.6 we can now estimate the order of $\|\Delta \bar{X}^p \Delta \bar{Z}^p\|$, and hence, using (5.7), we can estimate the predictor steplength t_k .

LEMMA 5.7. *We have*

$$\|\Delta \bar{X}^p \Delta \bar{Z}^p\| = O(\mu(\mu + \delta(X, Z))).$$

Proof. Combining Lemma 5.6 with Lemma 5.2, we have

$$(5.12) \quad \left\| \sqrt{\mu} I + \Delta \bar{X}^p - \hat{X}^a \right\| + \left\| \sqrt{\mu} I + \Delta \bar{Z}^p - \hat{Z}^a \right\| = O(\sqrt{\mu}(\mu + \delta(X, Z))),$$

so that, using (4.5),

$$(5.13) \quad \left\| \sqrt{\mu} I - \hat{X}^a \right\| + \left\| \sqrt{\mu} I - \hat{Z}^a \right\| = O(\sqrt{\mu}).$$

Moreover,

$$\begin{aligned}\Delta \bar{X}^p \Delta \bar{Z}^p &= (\hat{X}^a - \sqrt{\mu}I)(\hat{Z}^a - \sqrt{\mu}I) + (\hat{X}^a - \sqrt{\mu}I)(\sqrt{\mu}I + \Delta \bar{Z}^p - \hat{Z}^a) \\ &\quad + (\sqrt{\mu}I + \Delta \bar{X}^p - \hat{X}^a)\Delta \bar{Z}^p.\end{aligned}$$

Applying (5.8), (5.12), (5.13), and (4.5) to the above relation yields

$$\|\Delta \bar{X}^p \Delta \bar{Z}^p\| = O(\mu(\mu + \delta(X, Z))). \quad \square$$

THEOREM 5.8. *The iterates (X^k, Z^k) generated by Algorithm SDP(ϵ) converge to (X^a, Z^a) superlinearly with order $2/(1 + 2^{-r})$. The duality gap μ^k converges to zero at the same rate.*

Proof. From (5.7) we see that for any $t \geq 0$ satisfying

$$\beta_{k-1} + \|\Delta \bar{X}^p \Delta \bar{Z}^p\|_F / \mu_k \leq (1-t)((1-t)\mu_k/\epsilon)^{2^{-r}},$$

there holds

$$\delta(X + t\Delta X^p, Z + t\Delta Z^p) \leq ((1-t)\mu/\epsilon)^{2^{-r}}.$$

This implies using (4.9) and Lemma 5.7 that

$$\begin{aligned}(1-t_k)^{1+2^{-r}} &\leq (\beta_{k-1} + \|\Delta \bar{X}^p \Delta \bar{Z}^p\|_F / \mu_k)(\mu_k/\epsilon)^{-2^{-r}} \\ &= O(\mu_k^{1-2^{-r}}),\end{aligned}$$

so that

$$\mu_{k+1} = (1-t_k)\mu_k = O(\mu_k^{2/(1+2^{-r})}).$$

This shows that the duality gap converges to zero superlinearly with order $2/(1+2^{-r})$. It remains to prove that the iterates converge to the analytic center with the same order. Notice that

$$\begin{aligned}\|X^k - X(\mu_k)\|_F^2 &= \text{tr}(X^k - X(\mu_k))L^{-T}(L^T L)L^{-1}(X^k - X(\mu_k)) \\ &\leq \|L^T L\| \cdot \text{tr}(X^k - X(\mu_k))L^{-T}L^{-1}(X^k - X(\mu_k)) \\ (5.14) \quad &= \|L^T L\| \cdot \text{tr} L^{-1}(X^k - X(\mu_k))(X^k - X(\mu_k))L^{-T} \\ &\leq \|L^T L\|^2 \cdot \|L^{-1}(X^k - X(\mu_k))L^{-T}\|_F^2.\end{aligned}$$

However, using the definition of the F -norm and applying Lemma 5.4,

$$\|L^T L\|_F = \|LL^T\|_F = \|L_{\mu_k} \bar{D} L_{\mu_k}^T\|_F = O(\|L_{\mu_k} L_{\mu_k}^T\|_F).$$

Recall that $L_{\mu_k} L_{\mu_k}^T = \frac{1}{\sqrt{\mu_k}} X(\mu_k)$ by definition, so that using Lemma 3.1,

$$(5.15) \quad \|L^T L\|_F = O\left(\frac{1}{\sqrt{\mu_k}}\right).$$

Combining (5.14) and (5.15) with Lemma 5.3, we have

$$\|X^k - X(\mu_k)\|_F = O\left(\frac{1}{\sqrt{\mu_k}} \|L^{-1}(X^k - X(\mu_k))L^{-T}\|_F\right) = O(\delta(X^k, Z^k)) = O(\mu_k).$$

Hence, we obtain from Theorem 3.5 that

$$\|X^k - X^a\|_F = O(\mu_k).$$

Similarly, it can be shown that

$$\|Z^k - Z^a\|_F = O(\mu_k).$$

This shows that the iterates converge to the analytic center R -superlinearly, with the same order as μ_k converges to zero. $\square$

6. Conclusions. We have shown the global and superlinear convergence of the predictor-corrector algorithm SDP, assuming only the existence of a strictly complementary solution pair. The local convergence analysis is based on Theorem 3.5, which states that $\|X(\mu) - X^a\| + \|Z(\mu) - Z^a\| = O(\mu)$. By enforcing $\delta(X^k, Z^k) \rightarrow 0$, the iterates “inherit” this property of the central path. For the generalization of the Mizuno–Todd–Ye predictor-corrector algorithm in [17], we do not enforce $\delta(X^k, Z^k) \rightarrow 0$, and hence we cannot conclude superlinear convergence for it yet. In this respect, it will be interesting to study the asymptotic behavior of the corrector steps. Finally, it is likely that our line of argument can be applied to the infeasible primal-dual path following algorithms of Kojima, Shindoh, and Hara [8] and Potra and Sheng [16].

Acknowledgments. We wish to thank the referees for providing careful comments which led to substantial improvement in the presentation of the paper.

REFERENCES

- [1] F. ALIZADEH, *Interior point methods in semidefinite programming with applications to combinatorial optimization problems*, SIAM J. Optim., 5 (1995), pp. 13–51.
- [2] F. ALIZADEH, J. A. HAEBERLY, AND M. OVERTON, *Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results*, SIAM J. Optim., 8 (1998), to appear.
- [3] O. GÜLER, *Limiting behavior of weighted central paths in linear programming*, Math. Programming, 65 (1994), pp. 347–363.
- [4] C. HELMBERG, F. RENDL, R. J. VANDERBEI, AND H. WOLKOWICZ, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.
- [5] A. J. HOFFMAN, *On approximate solutions of systems of linear inequalities*, J. Res. National Bureau of Standards, 49 (1952), pp. 263–265.
- [6] M. KOJIMA, N. MEGIDDO, T. NOMA, AND A. YOSHISE, *A Unified Approach to Interior Point Algorithms for Linear Complementarity Problems*, Springer-Verlag, Berlin, 1991.
- [7] M. KOJIMA, M. SHIDA, AND S. SHINDOH, *Local convergence of predictor-corrector infeasible-interior-point algorithms for SDPs and SDLCPs*, Research Reports on Information Sciences, B-306, Department of Information Sciences, Tokyo Institute of Technology, 2-12-1 Oh-Okayama, Meguro-ku, Tokyo 152, Japan, 1995.
- [8] M. KOJIMA, S. SHINDOH, AND S. HARA, *Interior-point methods for the monotone linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.
- [9] C. J. LIN AND R. SAIGAL, *An Infeasible Start Predictor Corrector Method for Semi-Definite Linear Programming*, Tech. report, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI, 1995.
- [10] N. MEGIDDO, *Pathways to the optimal solution set in linear programming*, in Progress in Mathematical Programming, Interior-Point and Related Methods, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 131–158.
- [11] S. MIZUNO, M. J. TODD, AND Y. YE, *On adaptive-step primal-dual interior-point algorithms for linear programming*, Math. Oper. Res., 18 (1993), pp. 964–981.
- [12] R. D. C. MONTEIRO, *Primal-dual path-following algorithms for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 663–678.

- [13] R. D. C. MONTEIRO AND T. TSUCHIYA, *Limiting behavior of the derivatives of certain trajectories associated with a monotone horizontal linear complementarity problem*, Math. Oper. Res., 21 (1996), pp. 793–814.
- [14] Y. NESTEROV AND M. J. TODD, *Self-scaled barriers and interior-point methods for convex programming*, Math. Oper. Res., 22 (1997), pp. 1–42.
- [15] Y. NESTEROV AND M. J. TODD, *Primal-dual interior-point methods for self-scaled cones*, Tech. report 1125, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY, 1995.
- [16] F. A. POTRA AND R. SHENG, *A Superlinearly Convergent Primal-Dual Infeasible-Interior-Point Algorithm for Semidefinite Programming*, Report on Computational Mathematics 78, Department of Mathematics, University of Iowa, Iowa City, IA, 1995.
- [17] J. F. STURM AND S. ZHANG, *Symmetric Primal-Dual Path Following Algorithms for Semidefinite Programming*, Tech. report 9554/A, Econometric Institute, Erasmus University, Rotterdam, The Netherlands, 1995.
- [18] L. VANDENBERGHE AND S. BOYD, *A primal-dual potential reduction method for problems involving matrix inequalities*, Math. Programming, 69 (1995), pp. 205–236.
- [19] L. VANDENBERGHE AND S. BOYD, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.
- [20] Y. YE AND K. ANSTREICHER, *On quadratic and $O(\sqrt{n}L)$ convergence of a predictor-corrector algorithm for LCP*, Math. Programming, 62 (1993), pp. 537–551.
- [21] Y. ZHANG, *On extending primal-dual interior-point algorithms from linear programming to semidefinite programming*, SIAM J. Optim., 8 (1998), to appear.

RECONSTRUCTION WITH NOISY DATA: AN APPROACH VIA EIGENVALUE OPTIMIZATION*

DOMINIKUS NOLL[†]

Abstract. We present a nonlinear inverse filtering approach to problems such as power spectrum estimation of stationary time series or deconvolution of a blurred image. The technique is based on eigenvalue optimization and a numerical treatment may therefore be obtained using primal-dual interior-point methods for semidefinite programming.

Key words. inverse problems, power spectrum estimation, image restoration, Fisher information, eigenvalue optimization, semidefinite programming, large scale problems, finite-element discretization

AMS subject classifications. Primary, 49N50; Secondary, 90C06

PII. S105262349629856X

1. Introduction. In this paper we discuss a nonlinear inverse filtering approach to inverse problems such as the restoration of blurred images or the estimation of power spectra of stationary time series. In many situations, these inverse problems are difficult to treat due to a sizable imperfection of the available data.

Let us consider the problem of estimating an unknown nonnegative function $u(x) \geq 0$ on some domain $\Omega \subset \mathbb{R}^n$, given a finite number of measurements

$$(1.1) \quad \int_{\Omega} a_k(x) u(x) dx = b_k, \quad k = 0, \dots, m,$$

with $a_0(x), \dots, a_m(x)$ a given set of weight functions. This problem, with only a finite set of data available, is clearly underdetermined. More seriously, however, the data b_k may suffer from measurement errors and may make inversion of (1.1) a difficult task. As an example for such behavior consider the following.

Example 1.1 (power spectrum estimation). Here we wish to reconstruct the spectral density $u(x) \geq 0$, $x \in (-\pi, \pi)$ of a real stationary time series (X_t) , based on knowledge of the first $m + 1$ sample autocovariances $\hat{b}_0, \dots, \hat{b}_m$, which have been obtained from a realization $x_0, \dots, x_N$ of (X_t) (with $m \ll N$), using the statistic

$$\hat{b}_k = \frac{1}{N} \sum_{t=0}^{N-k} (x_t - \bar{x})(x_{t+k} - \bar{x}), \quad k = 0, \dots, m.$$

Equivalently (cf. [8]), we have to exhibit $u(x) \geq 0$ satisfying

$$(1.2) \quad \int_{-\pi}^{\pi} \cos kx u(x) dx = \hat{b}_k, \quad k = 0, \dots, m,$$

which is precisely a problem of type (1.1). It is well known (cf. [8]) that the sample autocovariances $\hat{b}_k$ are increasingly unstable for large k , and that a direct inversion of (1.2) based on the Fourier series of $u(x)$ usually fails. $\square$

*Received by the editors January 31, 1996; accepted for publication (in revised form) June 12, 1996.

<http://www.siam.org/journals/siopt/8-1/29856.html>

[†]Université Paul Sabatier, Laboratoire Approximation et Optimisation, 118 route de Narbonne, 31062 Toulouse Cedex, France (noll@dumbo.ups-tlse.fr).

As a second example, let us consider the following.

Example 1.2. (image restoration) Consider an image, represented by its distribution of gray levels $u(x) \geq 0$ over a region $\Omega \subset \mathbb{R}^2$. Suppose that during transmission through a news channel, the original image has suffered from various sources of degradation, some of them being signal dependent, some caused by random pixel noise. Assuming a linear model, the observed image $v(x) \geq 0$ may be represented by

$$(1.3) \quad v(x) = \int_{\Omega} q(x-y) u(y) dy + e(x), \quad x \in \Omega,$$

where the convolutional term is responsible for the signal dependent blurring ($q(x)$ being referred to as the *point spread function* or simply the *mask*), and where $e(x)$ represents the signal independent random noise. Suppose now that the recorded image $v(x)$ has been sampled at the nodes $x_1, \dots, x_m$ of a rectangular grid on Ω . Then the problem of deconvolving (1.3) will fit with the scheme (1.1), with the weight functions being $a_0 \equiv 1$, $a_k(x) = q(x_k - x)$, and the data $b_0 = 1$, $b_k = v(x_k)$. However, if the noise contribution $e(x)$ becomes sizable, it may be necessary to relax equation (1.1), $Au = b$, $\int_{\Omega} u dx = 1$, replacing it with a *tolerance model* of the form

$$(1.4) \quad |Au - b| \leq \epsilon, \quad \int_{\Omega} u(x) dx = 1,$$

where $b = (b_1, \dots, b_m)$, $Au = (\int_{\Omega} a_1 u dx, \dots, \int_{\Omega} a_m u dx)$, and $|\cdot|$ denotes the Euclidean norm on $\mathbb{R}^m$. Various strategies for selecting an appropriate tolerance ϵ are known. For instance, if the noise variance σ^2 is known—a hypothesis which is often met in practice—the strong law of large numbers implies

$$\frac{1}{m} \sum_{i=1}^m e_i^2 \rightarrow \sigma^2 \quad (m \rightarrow \infty),$$

which in view of $|Ax - b|^2 = |e|^2 = \sum_{i=1}^m e_i^2 \approx m\sigma^2$ suggests the choice $\epsilon = \sigma\sqrt{m}$ (see [31, 32] for a detailed discussion). $\square$

A successful idea to stabilize an inversion problem of type (1.1) or (1.4) in the presence of noisy data b_k is to introduce a performance index

$$\mathcal{I}(u) = \int_{\Omega} h(x, u(x), \nabla u(x)) dx$$

related to the problem under discussion, and to accept as a valid reconstruction the unique $u(x) \geq 0$ that minimizes $\mathcal{I}(u)$ under the constraints (1.1) (resp., (1.4)). Naturally, the strategy for designing $\mathcal{I}(u)$ should be justified by both theoretical models and performance in practice. While many different indices $\mathcal{I}(u)$ are being used in practice, a common feature seems to be their convexity.

In the power spectrum estimation problem (1.2), an objective often used in practice is the Burg entropy $\mathcal{I}(u) = -\int_{\Omega} \log u(x) dx$, which leads to a fast and, surprisingly, linear inverse method (see [7, 27]). For a theoretic justification see, e.g., [30]. By contrast, a nonlinear inverse filter is obtained when the Boltzmann–Shannon entropy/information measure $\mathcal{I}(u) = \int_{\Omega} u(x) \log u(x) dx$ is used, the discussion about the preference between the two choices being controversial; cf. [24].

The Boltzmann–Shannon entropy has also been tested for the image restoration problem (1.4), leading to a nonlinear inverse filter whose performance was found to

be significantly superior to more standard linear inverse filters used for this type of problems; cf. [31, 32, 22, 37].

One such linear inverse filter for the image restoration problem (1.4) is based on the performance index $\mathcal{I}(u) = \int_{\Omega} |\Delta u(x)|^2 dx$. For this model, a fast algorithmic solution using the 2D fast Fourier transform was presented by Hunt [23]. Other inverse filters could be based on inverse Wiener or Kalman filtering, but these methods require knowledge of the cross spectrum of the true image and the noise, and knowledge of the dynamics of the input process, two pieces of information which typically would be lacking in practical situations; cf. [2, 26].

In [36, 5, 6], a nonlinear inverse method for (1.1) based on the *Fisher information*

$$(1.5) \quad \mathcal{I}(u) = \int_{\Omega} \frac{|\nabla u(x)|^2}{u(x)} dx$$

was proposed. In the one-dimensional setting, sufficiently fast algorithms for the power spectrum estimation problem (1.2) and for problems with polynomial weights have been presented in [6, 7]. These techniques, however, are not suited for higher dimensions, and so far a successful numerical approach in higher dimensions was lacking. The main issue of the present paper is to present one such approach based on techniques from eigenvalue optimization. Notice that it seems highly desirable to have codes based on the Fisher information index (1.5), in particular for higher dimensional problems like (1.4), since Fisher information was found to perform well in the one-dimensional case and in particular was seen to outperform the Boltzmann–Shannon objective in quite a number of situations, as has been reported in [6, 7].

The purpose of the present paper is to analyze the restoration programs (1.5), (1.1) (resp., (1.5), (1.4)) using convex programming duality. This leads to their reformulation as eigenvalue optimization (resp., semidefinite programming) problems, the main idea being to postpone discretization to the latest possible moment. While sections 2 and 3 present the details of the analysis, section 4 gives a convergence result for the finite element discretization. The final section 5 presents some numerical experiments.

Our notation in sections 2 and 3 follows standard references on variational problems and Sobolev spaces like [14, 1, 39]. Readers who find the technical arguments in sections 2 and 3 (leading to the central Theorems 3.4, 3.5) too cumbersome to follow in full detail are invited to obtain these dual programming formulations by formally applying convex duality to the Lagrangian (3.1). This gives the correct answers right away, and reading the following sections 4 and 5 is still possible.

Let us close this introductory section with a typical application for using the Fisher information as a performance index. For further motivation see [36] or [5, App. I].

Example 1.3 (robustness and higher order statistics (see [40])). Consider the problem of estimating the parameters a_i, b_j of a linear input-output ARMA system of the form

$$y_t + a_1 y_{t-1} + \cdots + a_m y_{t-m} = x_t + b_1 x_{t-1} + \cdots + b_n x_{t-n},$$

where $t \in \mathbb{Z}$, x_t is the input, and y_t the output sequence, and where the orders n, m are known. Suppose that the input is white noise with mean zero, variance σ^2 , and probability density p .

Given a sample $y_1, \dots, y_N$ of outputs, the idea of maximum likelihood estimation of $\theta = (a_1, \dots, a_m, b_1, \dots, b_n)$ is to choose it to maximize a likelihood function of the

form

$$L(\theta) = - \sum_{t=1}^N \ell(r_t(\theta)),$$

where $r_i(\theta) = y_i - x_i - b_1 x_{i-1} - \dots - b_n x_{i-n} + a_1 y_{i-1} + \dots + a_m y_{i-m}$ is the i th residual.

How to find the appropriate $\ell(r)$? This could be based on the following observation, essentially made already by Fisher [17]; see also Doob [15]. Under some mild assumptions on the input distribution p and the smoothness of ℓ , the estimates $\hat{\theta}_N$ based on ℓ are asymptotically normally distributed, i.e.,

$$\sqrt{N}(\hat{\theta}_N - \theta^*) \sim \mathcal{N}(0, \alpha A / \sigma^2) \quad \text{as } N \rightarrow \infty.$$

Here the positive definite $(n+m) \times (n+m)$ matrix $A = A(\theta^*)$ depends only on the true parameter vector θ^* , while the scaling factors $\sigma^2 = \sigma^2(p)$ and $\alpha = \alpha(\ell, p)$ depend on the choice of ℓ and the underlying density p of the input law:

$$\alpha(\ell, p) = \int \ell'^2 p \, dx \Big/ \left(\int \ell'' p \, dx \right)^2$$

and could therefore be influenced by our choice of ℓ .

Let us assume in the following that the elements p of the class $\mathcal{P}$ have identical variance $\sigma_0^2 = \sigma^2(p)$. Designing a robust estimate now leads us to consider the worst case p_0 among the probability densities $p \in \mathcal{P}$ and then improve the asymptotic efficiency of the estimate $\hat{\theta}_N$ by choosing ℓ to minimize $\sup_p \alpha(\ell, p) = \alpha(\ell, p_0)$. That is, solve the minimax problem

$$\inf_{\ell} \sup_{p \in \mathcal{P}} \alpha(\ell, p).$$

Switching the infimum and the supremum, the inner minimization (now over the ℓ for a fixed p) admits a solution of the form $\ell = -\rho \log p$ (ρ constant). As $\alpha(\ell, p) = \alpha(\frac{1}{\rho} \ell, p)$, the problem is now equivalent to maximizing $\alpha(-\log p, p) = \mathcal{I}(p)^{-1}$, or equivalently, minimizing $\mathcal{I}(p)$, the Fisher information, over the admissible family $\mathcal{P}$ of probability densities.

The classical choice is of course $\ell_0(r) = r^2$. This is explained by choosing as the family $\mathcal{P}$ of admissible input distributions all laws p with given moments up to order 2. Indeed, the solution p_0 to the problem of minimizing the Fisher information $\mathcal{I}(p)$ subject to the constraints $\int p(x) \, dx = 1$, $\int p(x)x \, dx = 0$, and $\int p(x)x^2 \, dx = \sigma^2$ is just the Gaussian with mean zero and variance σ^2 , which leads to $\ell_0(r) = -\log p_0(r) = \mathcal{O}(r^2)$.

Suppose now that the white noise processes x_t has higher-order moments, say, up to order $m > 2$, m even. In this case, the classical choice $\ell_0(r) = r^2$ should be replaced by $\ell_0 = -\log p_0$, where p_0 now minimizes Fisher information over the set $\mathcal{P}$ of densities p with given moments up to order m . The results presented by Zivovnjovic [40] suggest that this choice of ℓ_0 will indeed improve the asymptotic efficiency of the maximum likelihood approach. The maximization of $L(\theta)$ would then, with no explicit ℓ_0 available, have to be performed numerically. $\square$

2. Variational formulation. In this section we will analyze the Fisher variational problem. First observe that the precise definition of the Fisher integrand $h(u, p)$

is

$$(2.1) \quad h(u, p) = \begin{cases} |p|^2/u & \text{if } u > 0, \\ 0 & \text{if } u = 0 \text{ and } p = 0, \\ +\infty & \text{otherwise,} \end{cases}$$

which is a normal convex integrand in the sense of Rockafellar [35]. The variational problem we consider is now

$$(P) \quad \begin{aligned} &\text{minimize} && \mathcal{I}(u) = \int_{\Omega} h(u(x), \nabla u(x)) dx = \int_{\Omega} \frac{|\nabla u(x)|^2}{u(x)} dx \\ &\text{subject to} && \int_{\Omega} a_k(x) u(x) dx = b_k, \quad k = 0, \dots, m, \\ &&& u(x) \geq 0, \quad u \in W^{1,1}(\Omega). \end{aligned}$$

In tandem with (P) and motivated by the image restoration problem (1.4) we also consider the relaxed Fisher program

$$(P_r) \quad \begin{aligned} &\text{minimize} && \mathcal{I}(u) = \int_{\Omega} \frac{|\nabla u(x)|^2}{u(x)} dx \\ &\text{subject to} && u(x) \geq 0, \quad u \in W^{1,1}(\Omega), \\ &&& \int_{\Omega} a_0(x) u(x) dx = b_0, \\ &&& |Au - b| \leq \epsilon, \end{aligned}$$

where $Au = (\int_{\Omega} a_1 u dx, \dots, \int_{\Omega} a_m u dx)$ and $b = (b_1, \dots, b_m)$. Notice that (P) and (P_r) are convex programs.

Throughout the following we shall assume that programs (P), (P_r) are feasible. Moreover, to avoid pathological situations, we shall assume throughout that $u \equiv 0$ is not an optimal solution of (P), (P_r) . Let us introduce the substitution

$$(2.2) \quad u(x) = v(x)^2, \quad 2v(x)\nabla v(x) = \nabla u(x),$$

which turns problems (P), (P_r) into the equivalent but nonconvex programs $(\tilde{P})$, $(\tilde{P}_r)$:

$$(\tilde{P}) \quad \begin{aligned} &\text{minimize} && \mathcal{J}(v) = 4 \int_{\Omega} |\nabla v(x)|^2 dx \\ &\text{subject to} && v \in W^{1,2}(\Omega), \\ &&& \int_{\Omega} a_k(x) v(x)^2 dx = b_k, \quad k = 0, \dots, m, \end{aligned}$$

resp.,

$$(\tilde{P}_r) \quad \begin{aligned} &\text{minimize} && \mathcal{J}(v) = 4 \int_{\Omega} |\nabla v(x)|^2 dx \\ &\text{subject to} && v \in W^{1,2}(\Omega), \quad |Av^2 - b| \leq \epsilon, \\ &&& \int_{\Omega} a_0(x) v(x)^2 dx = b_0. \end{aligned}$$

We are now in the position to prove existence and unicity of solutions of (P) and (P_r) .

PROPOSITION 2.1. *Let $a_0, \dots, a_m \in \mathcal{L}^\infty(\Omega)$, and assume $\int_{\Omega} a_0(x) dx \neq 0$. Let Ω be a bounded domain of class $\mathcal{C}^{0,1}$. Then program (P) has a unique solution in $W^{1,1}(\Omega)$, and similarly, (P_r) has a unique solution in $W^{1,1}(\Omega)$.*

Proof. (1) We prove existence of a solution in the case of program (P_r) . Using the transform (2.2), we may consider $(\tilde{P}_r)$. Let $v_n \in W^{1,2}(\Omega)$ be a minimizing sequence for $(\tilde{P}_r)$,

$$(2.3) \quad \mathcal{J}(v_n) \rightarrow \min, \quad |Av_n^2 - b| \leq \epsilon, \quad \int_{\Omega} a_0 v_n^2 dx = b_0.$$

Observe that $|v_n|$ is as well a minimizing sequence, since $|\nabla v_n| = |\nabla |v_n||$ a.e. Hence we may assume $v_n \geq 0$.

The first condition in (2.3) implies the boundedness of ∇v_n in $\mathcal{L}^2$. Suppose v_n was not bounded in $W^{1,2}(\Omega)$. Then, since ∇v_n is bounded we may decompose $v_n = w_n + c_n$, where w_n is a bounded sequence in $W^{1,2}(\Omega)$, and the c_n are constants tending to ∞ . Indeed, let $c_n = \int_{\Omega} v_n dx$ and $w_n = v_n - c_n$. Then $\int_{\Omega} w_n dx = 0$ in tandem with the boundedness of $\|\nabla w_n\|_2 = \|\nabla v_n\|_2$ implies boundedness of w_n in $\mathcal{L}^2$ (using a Poincaré inequality like, for instance, Ziemer [39, Thm. 4.2.1]). But then

$$b_0 = \int_{\Omega} v_n^2 a_0 dx = c_n^2 \int_{\Omega} a_0 dx + \mathcal{O}(c_n),$$

which implies $\int_{\Omega} a_0 dx = 0$, a contradiction. This implies the boundedness of v_n in $W^{1,2}$. We select a weakly convergent subsequence, also noted v_n , i.e., $v_n \rightharpoonup v$. The convexity of $\mathcal{J}$ implies its weak lower semicontinuity, hence

$$\liminf \mathcal{J}(v_n) \geq \mathcal{J}(v),$$

and so if v turns out to be feasible, it must be a minimum for $(\tilde{P}_r)$.

The fact that Ω is of class $\mathcal{C}^{0,1}$ implies the compactness of the inclusion $W^{1,2}(\Omega) \rightarrow \mathcal{L}^2(\Omega)$; cf. [21]. Hence v_n has a norm convergent subsequence in $\mathcal{L}^2$, also noted v_n , i.e., $v_n \rightarrow v$ in $\mathcal{L}^2$. This implies $v_n^2 \rightarrow v^2$ in $\mathcal{L}^1$, since $\|v_n^2 - v^2\|_1 \leq \|v_n - v\|_2 \|v_n + v\|_2 \rightarrow 0$. It also implies $v \geq 0$. Now $\epsilon \geq |Av_n^2 - b| \rightarrow |Av^2 - b|$ and $b_0 = \int a_0 v_n^2 dx \rightarrow \int a_0 v^2 dx$ together imply the feasibility of v . With v a minimum for $(\tilde{P}_r)$, $u = v^2$ is a minimum for (P_r) .

(2) To prove unicity, we may return to the original program (P_r) . The argument is now verbally the same as for the one-dimensional case given in [5, Thm. 2.1]. Notice that by contrast the solution of $(\tilde{P}_r)$ is not unique, but a unique nonnegative solution exists. $\square$

In order to obtain more information on the optimal solution of (P) (resp., (P_r)), we shall need a Lagrange multiplier result. For this we recall the concept of a pseudo-Haar family which was introduced in [4]: Call $a_0, \dots, a_m$ pseudo-Haar if $a_0|A, \dots, a_m|A$ are linearly independent on any measurable set $A \subset \Omega$ with positive measure.

PROPOSITION 2.2. *Suppose that the weight functions $a_0, \dots, a_m$ are pseudo-Haar and of class $\mathcal{C}^{t-1}$ for some $t \geq 1$. Suppose that Ω is bounded and of class $\mathcal{C}^{t+1}$. Let $\bar{v} \in W^{1,2}(\Omega)$ be the unique nonnegative optimal solution of program $(\tilde{P})$ (resp., $(\tilde{P}_r)$). Then there exist real numbers $\bar{\lambda}_0, \dots, \bar{\lambda}_m$ such that $\bar{v}$ satisfies the linear elliptic PDE with Neumann boundary conditions:*

$$(2.4) \quad -\Delta \bar{v}(x) + \sum_{k=0}^m \bar{\lambda}_k a_k(x) \bar{v}(x) = 0 \quad \text{on } \Omega,$$

$$(2.5) \quad \frac{\partial \bar{v}(x)}{\partial n_{\Omega}} = 0 \quad \text{on } \partial \Omega.$$

Moreover, $\bar{v} \in W^{1+t,2}(\Omega)$, and in particular, for $t \geq k + \frac{n}{2} - 1$, we have $\bar{v} \in C^k(\bar{\Omega})$.

Proof. We give the proof in the case of program $(\tilde{P}_r)$. The argument for $(\tilde{P})$ is similar. Introducing an auxiliary variable $e \in \mathbb{R}^m$, program $(\tilde{P}_r)$ takes the equivalent form

$$\begin{aligned} \text{minimize} \quad & f(v, e) = \int_{\Omega} |\nabla v(x)|^2 dx \\ \text{subject to} \quad & g_0(v, e) = \int_{\Omega} a_0(x) v(x)^2 dx - b_0 = 0, \\ & g_i(v, e) = \int_{\Omega} a_i(x) v(x)^2 dx - b_i - e_i = 0, \quad i = 1, \dots, m, \\ & g_{m+1}(v, e) = \epsilon^2 - |e|^2 \geq 0. \end{aligned}$$

We claim that there exist Lagrange multipliers $\bar{\lambda}_0, \dots, \bar{\lambda}_m$ and $\bar{\lambda}_{m+1} \geq 0$ such that the optimal pair $(\bar{v}, \bar{e})$, with $\bar{e}_i = \int_{\Omega} a_i \bar{v}^2 dx - b_i$, satisfies

1. $\nabla_v f(\bar{v}, \bar{e}) + \sum_{k=0}^m \bar{\lambda}_k \nabla_v g_k(\bar{v}, \bar{e}) = 0$;
2. $-\bar{\lambda}_i - 2\bar{\lambda}_{m+1} \bar{e}_i = 0, \quad i = 1, \dots, m$;
3. $\bar{\lambda}_{m+1}(\epsilon^2 - |\bar{e}|^2) = 0$.

To prove this, we need a Lagrange multiplier result in a Banach space setting. For instance, the multiplier rule obtained by Ginsburg and Ioffe [20, Thm. 3.1] applies to the above program. (Notice that this result is obtained through techniques of nonsmooth analysis and needs no constraint qualification. In particular, the linear independence of the a_i is not needed for this argument.) Now let the associated Lagrangian be defined as

$$L(v, e; \mu, \lambda_0, \dots, \lambda_m, \lambda_{m+1}) = \mu f(v, e) + \sum_{k=0}^{m+1} \lambda_k g_k(v, e);$$

then the quoted result provides multipliers $\bar{\mu} \geq 0, \bar{\lambda}_0, \dots, \bar{\lambda}_m, \bar{\lambda}_{m+1} \geq 0$ not all zero such that $\bar{\lambda}_{m+1} g_{m+1}(\bar{v}, \bar{e}) = 0$ and such that

$$(2.6) \quad (0, 0) \in \partial_a L(\bar{v}, \bar{e}; \bar{\mu}, \dots, \bar{\lambda}_{m+1}).$$

Here ∂_a refers to the *approximate subdifferential* (in the sense of [20]) with respect to (v, e) . Notice, however, that the functions f, g_k are all Fréchet differentiable with respect to the variables v, e , so ∂_a coincides with the Fréchet derivative. Evaluating (2.6) therefore, along with the constraints, gives the following conditions:

- 1.' $\bar{\mu} \nabla_v f(\bar{v}, \bar{e}) + \sum_{k=0}^m \bar{\lambda}_k \nabla_v g_k(\bar{v}, \bar{e}) = 0$;
- 2.' $-\bar{\lambda}_i - 2\bar{\lambda}_{m+1} \bar{e}_i = 0, \quad i = 1, \dots, m$;
- 3.' $\bar{\lambda}_{m+1} g_{m+1}(\bar{v}, \bar{e}) = 0$.

This is almost what we wish to prove, and it remains to check that the case $\bar{\mu} = 0$ may be excluded, i.e., that the problem is normal. It is this part of the proof which requires our qualification hypothesis.

Suppose we had $\bar{\mu} = 0$. Then we must have $\bar{\lambda}_{m+1} > 0$. Indeed, by item 2', $\bar{\lambda}_{m+1} = 0$ implied $\bar{\lambda}_1 = \dots = \bar{\lambda}_m = 0$. Hence $\bar{\lambda}_0 \neq 0$. But item 1' now reads as $\bar{\lambda}_0 \int_{\Omega} a_0(x) \bar{v}(x) h(x) dx = 0$ for all test functions h , hence $a_0 \bar{v} = 0$, and since $\bar{v} \neq 0$ this contradicts the pseudo-Haar assumption. So we must have $\bar{\lambda}_{m+1} > 0$.

Consequently, item 1' with $\bar{\mu} = 0$ gives us $\sum_{k=1}^m \bar{\lambda}_k \nabla_v g_k(\bar{v}, \bar{e}) = 0$, and by the definition of the Fréchet derivative of the g_k with respect to v , this implies

$$\sum_{k=1}^m \bar{\lambda}_k \int_{\Omega} a_k(x) \bar{v}(x) h(x) dx = 0$$

for all test functions $h \in \mathcal{C}^1(\Omega)$. Hence $\sum_1^m \bar{\lambda}_k a_k \bar{v} \equiv 0$. Since the a_k are assumed pseudo-Haar, and since $\bar{v}$ does not vanish identically, we deduce $\bar{\lambda}_1 = \dots = \bar{\lambda}_m = 0$. Inserting this in item 2' gives $\bar{e}_1 = \dots = \bar{e}_m = 0$, as $\bar{\lambda}_{m+1}$ had been agreed to be strictly positive. But then the inequality constraint could not have been active, and by item 3', we had to have $\bar{\lambda}_{m+1} = 0$, a contradiction. This reasoning proves $\bar{\mu} > 0$, and we may assume $\bar{\mu} = 1$.

We now exploit item 1 in the appropriate sense. Observe that f, g_i are all Fréchet differentiable on $W^{1,2}(\Omega) \times \mathbb{R}^m$ with $\langle \nabla_v f(v, e), h \rangle_{W^{1,2}} = \langle \nabla v, \nabla h \rangle_{\mathcal{L}^2}$ and $\langle \nabla_v g_k(v, e), h \rangle_{W^{1,2}} = \langle a_k \bar{v}, h \rangle_{\mathcal{L}^2}$. This implies that $\bar{v}$ is a weak solution of the Neumann problem (2.4), (2.5). Applying a boundary regularity result like [21, Thms. 9.1.15, 9.1.17] implies that the solution $\bar{v}$ is in fact in $W^{1+t,2}(\Omega)$, and using Green's formula, the equation (2.4) with Neumann boundary condition (2.5) are satisfied.

Finally, observe that the Sobolev embedding theorem gives $W^{1+t,2}(\Omega) \subset \mathcal{C}^k(\bar{\Omega})$ whenever $t + 1 > k + n/2$ (cf. [21, Thm. 6.2.30]), and this implies the last statement. $\square$

Remarks. 1) For $\mathcal{C}^\infty$ weight functions a_i , the classical inner regularity theory implies that the solution $\bar{v}$ of $(\tilde{P})$ (resp., $(\tilde{P}_r)$) is of class $\mathcal{C}^\infty(\Omega)$; see [21].

2) For applications it seems more natural to consider $\mathcal{C}^{0,1}$ domains, which includes, in particular, polyhedral domains used for finite elements. Here at least in dimensions $n = 2, 3$, and for smooth a_i , the generalized solution is in $\mathcal{C}^\infty(\Omega) \cap \mathcal{C}^{0,\alpha}(\bar{\Omega})$ for some Hölder exponent $0 < \alpha < 1$, so (2.4) is satisfied classically, but (2.5) has to be interpreted in the variational sense. The Hölder boundary regularity here may be derived from Theorem 3.2 of Murthy and Stampacchia [29].

3) In the imaging case $n = 2$, choosing $t \geq 2$ implies that the solution $\bar{v}$ is of class $\mathcal{C}^1$. The equation (2.4) and the boundary condition (2.5) are then satisfied in the classical sense.

A case of particular interest is Example 1.3. Here $\Omega = \mathbb{R}$, and the weight functions are smooth but unbounded, so Proposition 2.2 does not apply directly. We have the following result, which is not the most general one but covers the type of applications motivated by Example 1.3.

COROLLARY 2.3. *Consider the problem (P) of minimizing Fisher information $\mathcal{I}(u)$ subject to the moment constraints*

$$\int_{-\infty}^{\infty} x^k u(x) dx = b_k, \quad k = 0, \dots, m.$$

Then if $m = 2r$ is even, problem (P) has a unique solution $\bar{u}$ which is positive on $\mathbb{R}$. Moreover, there exist multipliers $\bar{\lambda}_0, \dots, \bar{\lambda}_{2r}$ such that $\bar{v} = \bar{u}^{1/2}$ is analytic and satisfies the linear differential equation with boundary conditions

$$v''(x) + \sum_{k=0}^{2r} \bar{\lambda}_k x^k v(x) = 0, \quad v(\pm\infty) = 0.$$

Proof. 1) We first prove existence. As before, we solve the transformed problem $(\tilde{P})$. Consider the weighted Sobolev space $\mathcal{W}$ of functions $v \in W^{1,2}(\mathbb{R})$ satisfying $\int v^2(x) x^{2r} dx < +\infty$. A corresponding Hilbert space norm is

$$\|v\| = \left(\int_{-\infty}^{\infty} v'(x)^2 dx + \int_{-\infty}^{\infty} v(x)^2 x^{2r} dx \right)^{1/2}.$$

Reasoning as in Proposition 2.1, consider a minimizing sequence v_n for $(\tilde{P})$. Then $\|v'_n\|_2 = \mathcal{O}(1)$ is clear. Now $\int x^{2r} v_n(x)^2 dx = b_{2r}$ implies the boundedness of v_n in $\mathcal{W}$. It is this part of the proof which requires m to be even. Selecting a weakly convergent subsequence, the rest of the proof remains unchanged.

2) The second part of the argument leading to the Euler equation is now exactly the same as in Proposition 2.2. Notice, however, that the elements of $\mathcal{W}$ vanish at infinity, so the boundary value problem is now actually a Dirichlet problem, which explains the boundary condition $\bar{v}(\pm\infty) = 0$. As the weight functions are analytic, the same must be true for the solution $\bar{v}$.

3) It remains to argue that $\bar{v}$ is positive on $\mathbb{R}$. This may be shown by reasoning similar to [5, Thm. 3.2(1)]. $\square$

Remark. Notice that for m odd, the Fisher problem with moment constraints up to order m in general fails to have a solution. In the case $m = 1$, this may be easily seen, e.g., for $b_0 = 1, b_1 = 0$ by constructing a sequence of symmetric $u_n > 0$ satisfying $\int u_n dx = 1$ and $\mathcal{I}(u_n) \rightarrow 0$. By symmetry, one always has $\int x u_n(x) dx = 0$.

3. Exploiting convexity. On transforming the programs $(P), (P_r)$ into the equivalent form $(\tilde{P}), (\tilde{P}_r)$, we were able to prove the existence of solutions and we obtained necessary optimality conditions using a Lagrange multiplier result. However, the convexity of the problems was not preserved by (2.2), so we lost some information, which turns out to be crucial for an appropriate numerical formulation. In the present section, we will therefore apply techniques from convex duality theory to obtain the missing information.

Introducing dummy variables p, e , program (P_r) takes the equivalent form

$$\begin{aligned} \text{minimize} \quad & \mathcal{I}(u, p) := \int_{\Omega} h(u(x), p(x)) dx = \int_{\Omega} \frac{|p(x)|^2}{u(x)} dx \\ \text{subject to} \quad & p = \nabla u, \quad Au - b - e = 0, \quad u \in W^{1,2}(\Omega), \\ & u(x) \geq 0, \quad \int_{\Omega} a_0(x) u(x) dx = b_0, \\ & \epsilon^2 - |e|^2 \geq 0. \end{aligned}$$

Recall that in section 2, the setting chosen for the programs $(P), (P_r)$ was $W^{1,1}(\Omega)$. However, as Proposition 2.2 shows the optimal solution to lie in a better space, restricting (P_r) to $W^{1,2}(\Omega)$ is justified and will provide a more convenient setting for the reasoning in this section.

The Lagrangian associated with this formulation is

$$(3.1) \quad \begin{aligned} L(u, p, e; w, \lambda, \mu, \nu) = & \mathcal{I}(u, p) + \langle w, \nabla u - p \rangle + \lambda \cdot (Au - b - e) \\ & + \mu \left(\int_{\Omega} a_0(x) u(x) dx - b_0 \right) + \nu(\epsilon^2 - |e|^2), \end{aligned}$$

where the state constraint $u \geq 0$ need not be modeled explicitly into the Lagrangian, since it is implicit in the definition of the Fisher functional $\mathcal{I}(u, p)$. With (3.1) program (P_r) takes the equivalent form

$$\begin{aligned} V(P_r) = \inf \Big\{ \sup \{ & L(u, p, e; w, \lambda, \mu, \nu) : w \in \mathcal{L}^2, \lambda \in \mathbb{R}^m, \mu \in \mathbb{R}, \nu \geq 0 \} : \\ & u \in W^{1,2}, p \in \mathcal{L}^2, |e| \leq \epsilon \Big\}. \end{aligned}$$

The dual program (D_r) associated with (P_r) is then defined as

$$V(D_r) = \sup \left\{ \inf_{u,p,e} L(u, p, e; w, \lambda, \mu, \nu) : w, \lambda, \mu, \nu \right\}$$

with the obvious relation $V(D_r) \leq V(P_r)$. Under the following Slater-type constraint qualification the two programs will be seen to be equivalent:

$$(CQ) \quad \begin{aligned} &\text{there exists } \hat{u} \in \mathcal{C}^1(\bar{\Omega}), \hat{u} > 0, \text{ such} \\ &\text{that } \int_{\Omega} a_0 \hat{u} dx = b_0 \text{ and } |A\hat{u} - b| < \epsilon. \end{aligned}$$

PROPOSITION 3.1. *Suppose that (CQ) holds for (P_r) , then there exist multipliers $\bar{w} \in \mathcal{L}^2(\Omega)$, $\bar{\lambda} \in \mathbb{R}^m$, and $\bar{\mu} \in \mathbb{R}$, $\bar{\nu} \geq 0$ such that*

$$V(D_r) = V(P_r) = \inf_{u,p,e} L(u, p, e; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu}) = L(\bar{u}, \nabla \bar{u}, \bar{e}; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu})$$

are satisfied. That is, $(\bar{u}, \nabla \bar{u}, \bar{e}; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu})$ is a saddle point of L in $W^{1,2}(\Omega) \times \mathcal{L}^2(\Omega) \times \mathbb{R}^m \times \mathcal{L}^2(\Omega) \times \mathbb{R}^m \times \mathbb{R} \times \mathbb{R}$.

Proof. The result is standard, so we only sketch the argument. Define a lower semicontinuous proper convex function $f : \mathcal{L}^2(\Omega) \times \mathbb{R}^m \times \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R} \cup \{+\infty\}$ by setting

$$f(z, \theta, \alpha, \sigma) = \inf \left\{ \mathcal{I}(u, p) : u \in W^{1,2}, p \in \mathcal{L}^2, \nabla u - p = z, Au - b - e = \theta, \right. \\ \left. \int_{\Omega} a_0 u dx - b_0 = \alpha, \epsilon^2 - |e|^2 \geq \sigma \right\}.$$

Then the value of (P_r) is $V(P_r) = f(0, 0, 0, 0) > -\infty$. It remains to show that $(0, 0, 0, 0)$ is in the interior of $\text{dom} f$ as a subset of $\mathcal{L}^2 \times \mathbb{R}^m \times \mathbb{R} \times \mathbb{R}$, for then $\partial f(0, 0, 0, 0)$ is nonempty and any $(-\bar{w}, -\bar{\lambda}, -\bar{\mu}, -\bar{\nu}) \in \partial f(0, 0, 0, 0)$ will provide the desired Lagrange multipliers. Now the constraint qualification (CQ) is seen to imply $f(z, \theta, \mu, \nu) < +\infty$ on a full neighborhood of the origin in $\mathcal{L}^2(\Omega) \times \mathbb{R}^m \times \mathbb{R} \times \mathbb{R}$, whence the result. $\square$

COROLLARY 3.2. *Suppose that (CQ) is satisfied for (P_r) and let $\bar{u}$ be its solution. Then $E = \{x \in \Omega : \bar{u}(x) = 0\}$ is a Lebesgue null set and the function $\log \bar{u}$ is in $W^{1,2}(\Omega)$. Moreover, $\bar{u} > 0$ on Ω if either (i) the dimension of the problem is $n = 1$ or (ii) the weight functions a_k are analytic on Ω .*

Proof. a) We first prove that $|\nabla \bar{u}|/\bar{u}$ is square integrable on $\Omega \setminus E = \{x \in \Omega : \bar{u}(x) > 0\}$. Since $(\bar{u}, \nabla \bar{u}, \bar{e}; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu})$ is a saddle point of L , we find that

$$0 \leq \frac{1}{t} (L(\bar{u} + t, \nabla \bar{u}, \bar{e}; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu}) - L(\bar{u}, \nabla \bar{u}, \bar{e}; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu})) \\ = \int_{\{\bar{u} > 0\}} \frac{-|\nabla \bar{u}|^2}{\bar{u}(\bar{u} + t)} dx + \int_{\Omega} (A^* \bar{\lambda} + \bar{\mu} a_0) dx < +\infty,$$

which uses the fact that $\nabla \bar{u} = 0$ a.e. on $\{\bar{u} = 0\}$. By convexity, the integrand is nondecreasing in $t > 0$, and hence monotone convergence allows us to pass to the limit $t \rightarrow 0$ under the integral sign. This implies $|\nabla \bar{u}|/\bar{u} \in \mathcal{L}^2(\Omega)$ on $\{\bar{u} > 0\}$.

b) We now argue that E is Lebesgue null. We first show how to derive this in dimension $n = 2$. Since $\bar{u} \in W^{1,2}(\Omega)$, [39, Thm. 2.1.4] implies that for almost all

$\xi \in \mathbb{R}$, $\bar{u}(\xi, \cdot)$ is absolutely continuous on $\Omega_\xi = \{\eta \in \mathbb{R} : (\xi, \eta) \in \Omega\}$. Moreover, by part a), $\partial_2 \bar{u}/\bar{u}$ is integrable on $\Omega \setminus E$, and hence for almost all ξ , $\partial_2 \bar{u}(\xi, \cdot)/\bar{u}(\xi, \cdot)$ is integrable on $\Omega_\xi \cap \{\eta : \bar{u}(\xi, \eta) > 0\}$. Fix some ξ satisfying both conditions. We claim that the following alternative holds: Either $\bar{u}(\xi, \cdot) > 0$ on Ω_ξ , or $\bar{u}(\xi, \cdot) = 0$ identically there. Indeed, suppose we had $\bar{u}(\xi, \eta) = 0$ for some $\eta \in \Omega_\xi$, but $\bar{u}(\xi, \cdot) > 0$ near the zero η , say on $I = (\eta, \eta')$. Then $\partial_2 \bar{u}(\xi, \cdot)/\bar{u}(\xi, \cdot)$ fails to be integrable on I , which is a contradiction. Indeed, for $\delta > 0$ we have

$$\int_{\eta+\delta}^{\eta'} \frac{\partial_2 \bar{u}(\xi, \eta)}{\bar{u}(\xi, \eta)} d\eta = \log \bar{u}(\xi, \eta') - \log \bar{u}(\xi, \eta + \delta) \rightarrow +\infty \quad (\delta \rightarrow 0).$$

This proves the claimed alternative.

Suppose now that $\bar{u}(\xi, \cdot) \equiv 0$ on some Ω_ξ . This shows that every $\bar{u}(\cdot, \eta)$ vanishes somewhere on $\Omega^\eta = \{\xi : (\xi, \eta) \in \Omega\}$. Repeating the above argument with respect to the second coordinate gives the same alternative as above, but with the first possibility now excluded. Hence for almost every η , we have $\bar{u}(\cdot, \eta) = 0$ identically, i.e., $\bar{u} \equiv 0$ a.e. This is impossible, and hence E must be a null set in dimension $n = 2$. For larger n we have to iterate the above argument using Fubini's theorem.

With $\bar{u} > 0$ a.e., it follows that $\log \bar{u}$ is defined a.e., and such that $\nabla(\log \bar{u}) \in L^2(\Omega)$ by part a). It follows via a Poincaré inequality that $\log \bar{u}$ is in $W^{1,2}(\Omega)$.

c) Statement (i) now follows, for in dimension $n = 1$ the embedding theorem gives $W^{1,2}(\Omega) \subset C(\bar{\Omega})$, and so $\log \bar{u}$ is continuous, which means $\bar{u} > 0$. As for statement (ii), observe that the differential operator in (2.4) has now analytic coefficients, so by Lewy's classical argument (cf. [12]) the solution $\bar{v}$, and hence $\bar{u} = \bar{v}^2$, must be analytic on Ω . For analytic $\bar{u}$, $\log \bar{u}$ could not be integrable in a neighborhood of a zero of $\bar{u}$, so $\bar{u} > 0$. $\square$

Remark. The above reasoning shows that the set E could not contain any line segments or even pieces of curves.

In the following, assume that (CQ) is satisfied with $\hat{u} \geq \eta > 0$ of class $C^1(\bar{\Omega})$, so that Proposition 3.1 applies. This implies

$$L(\hat{u} + u, \nabla \hat{u}, \hat{e}; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu}) \geq V(P_r) = V(D_r) > -\infty$$

for every $u \in C^1(\bar{\Omega})$ having $\|u\|_\infty \leq \eta/2$. Since $\mathcal{I}(\hat{u} + u, \nabla \hat{u})$ is bounded on $\|u\|_\infty \leq \eta/2$, the linear form $u \rightarrow \langle \nabla u, \bar{w} \rangle$ is bounded below (and hence bounded) on $\|u\|_\infty \leq \eta/2$. It extends to a bounded linear form on $C(\bar{\Omega})$. Hence there exists a signed Borel measure $\bar{m}$ on $\bar{\Omega}$ satisfying

$$-\langle u, \bar{m} \rangle = \langle \nabla u, \bar{w} \rangle$$

for every $u \in C^1(\bar{\Omega})$. Notice that $\bar{m} = \text{div } \bar{w}$ in the distributional sense, but $\bar{m}$ also comprises the singular boundary measure. Let $\bar{m}$ be decomposed according to Lebesgue, that is,

$$\bar{m} = \bar{m}_a + \bar{m}_s,$$

where $\bar{m}_a$ is absolutely continuous with respect to Lebesgue measure and $\bar{m}_s$ is singular. Let $d\bar{m}_a = \bar{\phi} dt$, then the Lagrangian (3.1) may be rewritten in the form

$$(3.2) \quad \begin{aligned} L(u, v, e; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu}) = & \mathcal{I}(u, v) - \langle \bar{\phi}, u \rangle - \langle \bar{m}_s, u \rangle - \langle \bar{w}, v \rangle + \langle A^t \bar{\lambda}, u \rangle \\ & - \bar{\lambda} \cdot e - \bar{\lambda} \cdot b + \langle \bar{\mu} a_0, u \rangle - \bar{\mu} b_0 + \bar{\nu}(\epsilon^2 - |e|^2). \end{aligned}$$

Notice that in contrast with (3.1), the presence of the term $\langle \bar{m}_s, u \rangle$ in formula (3.2) requires at least $u \in \mathcal{C}(\Omega)$. Assuming in the following that the program data allow for Proposition 2.2 to apply (with $t \geq n/2$), the optimal $\bar{u}$ is of class $\mathcal{C}^1(\bar{\Omega})$. In view of Proposition 3.1, this means that $(\bar{u}, \nabla \bar{u}, \bar{e}; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu})$ is now a saddle point of (3.2) in the duality $\mathcal{C}(\Omega) \times \mathcal{L}^2(\Omega) \times \mathbb{R}^m \times \mathcal{L}^2(\Omega) \times \mathbb{R}^m \times \mathbb{R} \times \mathbb{R}$.

Following the outline of convex duality theory, we shall now calculate (using (3.2)) the Fenchel conjugate $L^*(\cdot, \cdot, \cdot; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu})$ of $L(\cdot, \cdot, \cdot; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu})$ with respect to this duality. As $\bar{m}_s$ is concentrated on a Lebesgue null set, the supremum over the term $\langle \bar{m}_s, u \rangle$ may be controlled by a portion of each u living on a set with arbitrarily small Lebesgue measure, not affecting the supremum to be obtained for the remaining terms, (cf. [5] for a similar argument). The methods of Rockafellar [35] therefore lead to the following formula

$$L^*(y, z, \alpha; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu}) = \mathcal{I}^*(y + \bar{\phi} - \bar{\mu}a_0 - A^t \bar{\lambda}, z + \bar{w}) + \bar{\mu}b_0 + \bar{\lambda} \cdot b + \epsilon |\bar{\lambda} + \alpha|, \\ + \sup_{u \geq 0} \langle \bar{m}_s, u \rangle.$$

Here $A^t : \mathbb{R}^m \rightarrow \mathcal{L}^2(\Omega)$ denotes the adjoint of A , defined as $A^t \lambda = \sum_1^m \lambda_k a_k$. Now the conjugate $\mathcal{I}^*$ of the Fisher integral functional (1.5) is found to be $\mathcal{I}^*(y, z) = \int_{\Omega} h^*(y(x), z(x)) dx$, where

$$h^*(y, z) = \begin{cases} 0 & \text{if } y + \frac{1}{4}|z|^2 \leq 0, \\ +\infty & \text{otherwise} \end{cases}$$

and dealing with the singular term in L^* remains.

Observe that L^* has finite value somewhere; hence $\sup_{u \geq 0} \langle \bar{m}_s, u \rangle = \sup_{u \geq 0} (\langle \bar{m}_s^+, u \rangle - \langle \bar{m}_s^-, u \rangle)$ must be finite. This implies that the positive part $\bar{m}_s^+$ must vanish. The remaining term is $\sup_{u \geq 0} \langle -\bar{m}_s^-, u \rangle = 0$. Since $L^*(0, 0, 0; \cdot, \cdot, \cdot, \cdot)$ is finite at $(\bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu})$, we must have $\bar{\phi} - \bar{\mu}a_0 - A^t \bar{\lambda} + \frac{1}{4}|\bar{w}|^2 \leq 0$ a.e. Moreover, the saddle point condition (Proposition 3.1) implies

$$(3.3) \quad \langle \bar{u}, y \rangle + \langle \nabla \bar{u}, z \rangle + \bar{e} \cdot \alpha \leq \epsilon (|\bar{\lambda} + \alpha| - |\bar{\lambda}|)$$

for all (y, z, α) for which L^* is finite, or what is the same, for which

$$(3.4) \quad y + \bar{\phi} - \bar{\mu}a_0 - A^t \bar{\lambda} + \frac{1}{4}|z + \bar{w}|^2 \leq 0 \quad \text{a.e.}$$

Suppose now we have $\bar{\phi} - \bar{\mu}a_0 - A^t \bar{\lambda} + \frac{1}{4}|\bar{w}|^2 < -\theta < 0$, (for some $\theta > 0$) on a set Σ . Choosing $y = \theta \chi_{\Sigma}$ gives a tuple $(y, 0, 0)$ admitted in (3.4); hence $\langle \bar{u}, \theta \chi_{\Sigma} \rangle \leq 0$. As $\bar{u}$ is strictly positive, this is possible only when Σ is a null set. This reasoning shows that

$$(3.5) \quad \bar{\phi} - \bar{\mu}a_0 - A^t \bar{\lambda} + \frac{1}{4}|\bar{w}|^2 = 0 \quad \text{a.e.}$$

The feasibility condition (3.4) therefore becomes $y + \frac{1}{2}z \cdot \bar{w} + \frac{1}{4}|z|^2 \leq 0$ a.e. Consequently, on setting $y = -\frac{1}{2}z \cdot \bar{w} - \frac{1}{4}|z|^2$, we obtain a feasible $(y, z, 0)$. Inserting this in (3.3) gives

$$\left\langle \bar{u}, -\frac{1}{2}z \cdot \bar{w} \right\rangle - \frac{1}{4} \langle \bar{u}, |z|^2 \rangle + \langle \nabla \bar{u}, z \rangle \leq 0$$

for every $z \in \mathcal{L}^\infty(\Omega)$. As for small z the quadratic term becomes negligible, this readily implies

$$\left\langle -\frac{1}{2}\bar{u}\bar{w} + \nabla\bar{u}, z \right\rangle \leq 0$$

for every such z , and hence, on replacing z by $-z$, we obtain

$$(3.6) \quad \frac{1}{2}\bar{w} = \frac{\nabla\bar{u}}{\bar{u}} = \nabla(\log \bar{u}) \quad \text{a.e.}$$

In particular, $\bar{w}$ is seen to be a gradient field, and equation (3.5) with $\bar{\phi} = (\operatorname{div}\bar{w})_a = (\operatorname{div} \nabla(2 \log \bar{u}))_a$ now reads as

$$(3.7) \quad 2(\Delta(\log \bar{u}))_a + |\nabla(\log \bar{u})|^2 - \bar{\mu}a_0 - A^t\bar{\lambda} = 0.$$

The question which remains is when we may expect $\bar{m}_s^-$ to vanish. This is closely related to the question whether or not the implicit constraint $u \geq 0$ in our original problem (P) becomes active.

LEMMA 3.3. *The singular measure $\bar{m}_s^-$ is concentrated on the set $E = \{x \in \bar{\Omega} : \bar{u}(x) = 0\}$.*

Proof. We sketch the argument, which is of classical variational type. Observe that the infimum $V(P_r)$ of $L(\cdot, \cdot, \cdot; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu})$ is attained at $(\bar{u}, \nabla\bar{u}, \bar{e})$. Let $Q = \operatorname{support}(\bar{m}_s^-) \cap \{x : \bar{u}(x) > 0\}$. For a given $\epsilon > 0$ we find disjoint open intervals $I_1, \dots, I_r$ covering the support of $\bar{m}_s^-$ up to a set of $\bar{m}_s^-$ -measure $< \epsilon$, and such that the whole family has Lebesgue measure $< \epsilon$. Now start modifying the function $\bar{u}$ on the small set

$$Q_\epsilon = \bigcup_{i=1}^r I_i \cap \{x : \bar{u}(x) > 0\}$$

to build a function u_ϵ , which on a relevant portion P_ϵ of Q_ϵ (i.e., $\bar{m}_s^-(Q_\epsilon \setminus P_\epsilon) \rightarrow 0$) equals $\frac{1}{2}\bar{u}$, while outside Q_ϵ still equals $\bar{u}$. The construction gives

$$\langle \bar{m}_s^-, u_\epsilon \rangle \rightarrow \frac{1}{2} \langle \bar{m}_s^-, \bar{u} \rangle.$$

As $u_\epsilon \rightarrow \bar{u}$ in the sense of Lebesgue measure, the nonsingular terms in (3.2) are not affected, that is, on passing to the limit $\epsilon \rightarrow 0$, we find

$$V(P_r) \leq L(u_\epsilon, \nabla\bar{u}, \bar{e}; \bar{w}, \bar{\lambda}, \bar{\mu}, \bar{\nu}) \rightarrow V(P_r) - \frac{1}{2} \langle \bar{m}_s^-, \bar{u} \rangle,$$

giving $\langle \bar{m}_s^-, \bar{u} \rangle = 0$, and hence $\bar{m}_s^-(Q) = 0$, as claimed. $\square$

Remark. In view of Corollary 2.3, $\bar{m}_s = 0$ if either the dimension is $n = 1$ or the weights a_i are analytic functions on Ω .

Under our standing assumption on the program data, $\bar{u}$ is continuous. Then the exceptional set $E := \{x : \bar{u}(x) = 0\}$ is closed. Moreover, E is a Lebesgue null set by Corollary 3.2, and so $\Omega \setminus E$ is an open dense set. Now we compare equations (2.4) and (3.5). On $\Omega \setminus E$, the substitution $\frac{1}{2}\bar{w} = \nabla\bar{u}/\bar{u} = 2\nabla\bar{v}/\bar{v}$ may just be understood in the classical sense. The a_k being linearly independent, this implies that (up to a factor $\frac{1}{4}$) the multipliers $\bar{\lambda}_k$ coming from Proposition 2.2, and the multipliers λ_k coming from the convex duality of this section are identical (a fact which was to be expected, but

still had to be verified). Since by Lemma 3.3, $(\operatorname{div} \bar{w})_a = \operatorname{div} \bar{w}$ on $\Omega \setminus E$, equations (3.5) and (2.4) are equivalent on this set. We shall see how this piece of information may be used to obtain a convenient formulation of the dual program (∇_r) .

Recall that the dual (D_r) consists in maximizing $-L^*(0, 0, 0; w, \lambda, \mu, \nu) = -\mu b_0 - \sum_{k=1}^m \lambda_k b_k - \epsilon |\lambda|$ over those (w, λ, μ, ν) satisfying (3.5). On replacing (3.5) by (2.4), and taking into account that the boundary condition coming along with (2.4) is the Neumann condition (2.5), we obtain the following characterization of the dual (D_r) of (P_r) .

THEOREM 3.4. *Assuming (CQ) and the conditions on the program data from Proposition 2.2 (with $t \geq n/2$), the dual (D_r) of (P_r) takes the equivalent form*

$$\begin{aligned}
 (D_r) \quad & \text{maximize} \quad -\mu b_0 - \sum_{k=1}^m \lambda_k b_k - \epsilon |\lambda| \\
 & \text{subject to} \quad \Delta v = \frac{1}{4} \sum_{k=1}^m \lambda_k a_k(x) v + \frac{1}{4} \mu a_0(x) v \quad \text{on } \Omega, \\
 & \quad \frac{\partial v}{\partial n_\Omega} = 0 \quad \text{on } \partial\Omega, \\
 & \quad v \geq 0, \quad \int_{\Omega} a_0 v^2 dx = b_0.
 \end{aligned}$$

Moreover, the optimal solution $\bar{u}$ of (P_r) is obtained from the optimal solution $(\bar{\lambda}, \bar{\mu}, \bar{v})$ of (D_r) via $\bar{u} = \bar{v}^2$.

So far the arguments in this section have been for program (P_r) and its dual. Using a similar reasoning, we obtain the following result on program (P) .

THEOREM 3.5. *Suppose that the above assumptions on the program data are satisfied, but with the constraint qualification (CQ) for (P_r) replaced by the following:*

$$\text{There exists } \hat{u} \in C^1(\bar{\Omega}), \hat{u} > 0, \text{ such that } A\hat{u} = b.$$

Then the dual (D) of (P) has the equivalent form

$$\begin{aligned}
 (D) \quad & \text{maximize} \quad -\mu b_0 - \sum_{k=1}^m \lambda_k b_k \\
 & \text{subject to} \quad \Delta v = \frac{1}{4} \sum_{k=1}^m \lambda_k a_k v + \frac{1}{4} \mu a_0(x) v \quad \text{on } \Omega, \\
 & \quad \frac{\partial v}{\partial n_\Omega} = 0 \quad \text{on } \partial\Omega, \\
 & \quad v \geq 0, \quad \int_{\Omega} a_0 v^2 dx = b_0.
 \end{aligned}$$

Moreover, the optimal solution $\bar{u}$ of (P) may be recovered from the optimal solution $(\bar{\lambda}, \bar{\mu}, \bar{v})$ of (D) via $\bar{u} = \bar{v}^2$.

4. Numerical aspects. In this section we obtain a formulation of the programs (D) (resp., (D_r)) which is appropriate for a numerical treatment. We adopt the assumptions about the program data made in Theorems 3.4 (resp., 3.5), assuming in addition that $a_0 \geq 0$. We start by analyzing program (D) , which we reformulate slightly: replace λ_k by $-4\lambda_k$, $k = 1, \dots, m$, and μ by -4μ , and multiply the objective

with $1/4$. Then the dual (D) takes the equivalent form

$$\begin{aligned}
 \text{(D)} \quad & \text{maximize} \quad \mu b_0 + \sum_{k=1}^m \lambda_k b_k \\
 & \text{subject to} \quad \Delta v + \sum_{k=1}^m \lambda_k a_k(x) v + \mu a_0(x) v = 0 \quad \text{on } \Omega, \\
 & \quad \frac{\partial v}{\partial n_\Omega} = 0 \quad \text{on } \partial\Omega, \\
 & \quad v \geq 0, \quad \int_{\Omega} a_0 v^2 dx = b_0.
 \end{aligned}$$

This shows that for any feasible $(\mu, \lambda_1, \dots, \lambda_m, v)$, μ is an eigenvalue for the Neumann eigenvalue problem

$$\begin{aligned}
 \text{(N)} \quad & L_\lambda[v] + \mu a_0(x) v = 0 \quad \text{on } \Omega, \\
 & \frac{\partial v}{\partial n_\Omega} = 0 \quad \text{on } \partial\Omega
 \end{aligned}$$

with differential operator $L_\lambda[u] = \Delta u + (\sum_{k=1}^m \lambda_k a_k(x)) \cdot u$, and with v the corresponding eigenfunction, normalized to satisfy $\int_{\Omega} a_0 v^2 dx = b_0$. Now observe that an eigenfunction which does not change sign must belong to the smallest eigenvalue of (N), so any such μ must be the smallest eigenvalue for (N). Indeed, this is a classical fact for $a_0 \geq 0$, cf. [12, sect. 6], but may be extended to more general weights a_0 (see [9]).

Let us introduce some notation. For a self-adjoint second-order elliptic partial differential operator $L = L[\cdot]$, the smallest eigenvalue for the Neumann problem (N), which is known to be simple (cf. [21, 12]), will be denoted as $\mu_{\min}(L)$. The function $\mu_{\min}$ is known to be a concave function of the argument, see [28]. Since $\lambda \rightarrow L_\lambda[\cdot]$ is affine, we derive that $\lambda \rightarrow \mu_{\min}(L_\lambda[\cdot])$ is a concave function defined on the whole of $\mathbb{R}^m$. This function is even analytic, since the smallest eigenvalue of L_λ is simple (cf. [25, VII, Thm. 1.8] and also [12, 28]). The dual (D) may therefore be recast as the following finite-dimensional unconstrained and smooth concave program on $\mathbb{R}^m$, which is recognized as a semidefinite eigenvalue optimization program (in the sense of [28, 38]):

$$\text{(D)} \quad \text{maximize } f(\lambda) := \mu_{\min}(L_\lambda[\cdot]) b_0 + \sum_{k=1}^m \lambda_k b_k, \quad \lambda \in \mathbb{R}^m.$$

Given the optimal solution $\bar{\lambda}$ of (D), due to (3.6), the optimal solution $\bar{u}$ of the original program (P) is $\bar{u} = \bar{v}^2$, where $\bar{v}$ is the unique eigensolution belonging to the eigenvalue $\mu_{\min}(L_{\bar{\lambda}})$, which has been normalized to satisfy $\int_{\Omega} a_0 \bar{v}^2 dx = b_0$.

The dual (D_r) of (P_r) may be treated in the same way, leading to the concave unconstrained program

$$\text{(D}_r\text{)} \quad \text{maximize } g(\lambda) := \mu_{\min}(L_\lambda[\cdot]) b_0 + \sum_{k=1}^m \lambda_k b_k - \epsilon |\lambda|, \quad \lambda \in \mathbb{R}^m.$$

The return formula for the optimal solution $\bar{u}$ of (P_r) in terms of the optimal $\bar{\lambda}$ for (D_r) is exactly the same as for (P), (D) above.

The fact that the variational problem (P) is equivalent to a smooth concave unconstrained finite-dimensional optimization problem is by itself remarkable. It must

nevertheless be observed that the function $\lambda \rightarrow \mu_{\min}(L_\lambda)$ cannot be calculated explicitly, so for the numerical approach we are forced to replace the operator L_λ by a discretization L_λ^h . The error thereby introduced may be analyzed through the error term $|\mu_{\min}(L_\lambda) - \mu_{\min}(L_\lambda^h)|$.

Motivated by the image restoration problem (1.4), a natural discretization for (D_r) appears to be a finite difference approximation L_λ^h of L_λ on a rectangular grid Ω_h with mesh $h > 0$ on Ω . On the other hand, as our program (D) comprises a variety of eigenvalue optimization problems for second-order elliptic operators with a physical background, one might as well use a finite element or Rayleigh–Ritz discretization. (This was, for instance, chosen for the numerical treatment of the strongest column problem by Cox and Overton [13]). In the following we shall discuss the discretization errors associated with either method, and the sensitivity of the solution of (D) (resp., (P)) as compared to its discretized version.

LEMMA 4.1. *Let $L_\lambda[u] = \Delta u + \sum_{k=1}^m \lambda_k a_k(x)u$, and let $\phi(\lambda) = \mu_{\min}(L_\lambda[\cdot])$ be the minimal eigenvalue for the Neumann problem (N) above. Then ϕ is analytic and the first-order partial derivatives are*

$$(4.1) \quad \frac{\partial \phi(\lambda)}{\partial \lambda_i} = \int_{\Omega} a_i(x) e_\lambda(x)^2 dx,$$

where e_λ is the normalized eigenfunction belonging to the eigenvalue $\mu_{\min}(L_\lambda)$.

Proof. Since the smallest eigenvalue of L_λ is simple, the analyticity of ϕ is a direct consequence of Theorem VII. 1.8 in [25]. To prove (4.1) we shall need an approximation argument.

Let L_λ^h be a finite difference approximation for L_λ with mesh $h > 0$. For h small enough, the first eigenvalue $\phi_h(\lambda) := \mu_{\min}(L_\lambda^h)$ of the matrix L_λ^h is simple. Now observe that the eigenfunctions belonging to L_λ^h are defined on the Euclidean space $\mathcal{L}_h^2$ with weighted norm $|e|_0 = h^{n/2}(\sum_{x \in \Omega_h} e(x))^{1/2}$, so we have

$$(4.2) \quad \frac{\partial \phi_h(\lambda)}{\partial \lambda_i} = (h^{n/2} e_\lambda^h) \cdot \frac{\partial L_\lambda^h}{\partial \lambda_i} (h^{n/2} e_\lambda^h)$$

according to a well-known formula in finite dimensions (cf. [28]). Here the eigenvectors e_λ^h must satisfy $|e_\lambda^h|_0 = \|h^{n/2} e_\lambda^h\|_2 = 1$.

Now observe that $\partial L_\lambda^h / \partial \lambda_i$ is the diagonal matrix with entries $a_i(x)$, $x \in \Omega_h$, the discretization of $a_i(x)$. Therefore the right-hand side of (4.2) equals

$$\sum_{x \in \Omega_h} h^n a_i(x) e_\lambda^h(x)^2,$$

which converges to the right-hand side of (4.1) as $h \rightarrow 0$. (For details concerning the finite difference discretization see, e.g., [21, sect. 11.2]). $\square$

Let us now consider the case of a finite-element or Rayleigh–Ritz approximation of problems (D), (D_r) . Let $V = W^{1,2}(\Omega)$, $U = \mathcal{L}^2(\Omega)$ be the Gelfand pair associated with the Neumann problem (N), and let V_h be an approximating sequence of finite-dimensional subspaces of V . We assume that (cf. [21, sect. 11.2])

$$(4.3) \quad \text{dist}(u, V_h) \leq C \|u\|_{1+s} h^s \quad \text{for every } u \in W^{1+s,2}(\Omega),$$

which means that the basis functions are polynomials of degree $s \geq 1$ on the elements; cf. [11, Thm. 3.2.1]. Further, as in previous sections, we assume linear independence

and smoothness of the weight functions $a_0, a_1, \dots, a_m$, and for convenience we let $a_0 \equiv 1$. We now work on a bounded domain Ω with $\mathcal{C}^{0,1}$ boundary.

Let L_λ^h be the operator belonging to the approximation (N_h) of the variational problem (N) on the subspace V_h . Let $f(\lambda) = \mu_{\min}(L_\lambda)b_0 + \sum_1^m \lambda_k b_k$ be the objective of (D), and let the objective belonging to the approximate program (D_h) be $f_h(\lambda) = \mu_{\min}(L_\lambda^h)b_0 + \sum_1^m \lambda_k b_k$ to be maximized over $\mathbb{R}^m$.

Let the optimal solutions of (D), (D_h) be $\bar{\lambda}, \lambda^h$, respectively. Let $\bar{v}$ (resp., v^h) be the positive eigensolutions belonging to $\mu_{\min}(L_{\bar{\lambda}})$ (resp., $\mu_{\min}(L_{\lambda^h}^h)$), both normalized to satisfy $\int_\Omega \bar{v}^2 dx = b_0 = \int_\Omega (v^h)^2 dx$. Then we consider $u^h = (v^h)^2$ as an approximation of $\bar{u}$, and our interest will be in the error $\|\bar{u} - u^h\|_{W^{1,1}}$.

THEOREM 4.2. *Under the above assumptions on the program data, consider a finite element approximation V_h of V satisfying (4.3). Then we have the following statements.*

1. *For every compact set $\Lambda \subset \mathbb{R}^m$, there exists a constant C (depending only on s and Λ) such that the estimates*

$$|f(\lambda) - f_h(\lambda)| \leq Ch^{2s} \quad \text{and} \quad |\nabla f(\lambda) - \nabla f_h(\lambda)| \leq Ch^s$$

are satisfied for all $h > 0$ and $\lambda \in \Lambda$.

2. *The optimal solutions converge: $\lambda_h \rightarrow \bar{\lambda}$, $v^h \rightarrow \bar{v}$ (in $W^{1,2}$), and $u^h \rightarrow \bar{u}$ (in $W^{1,1}$), and the rate of convergence is $\|\bar{u} - u^h\|_{W^{1,1}} = \mathcal{O}(h^s)$.*

Proof. a) It is a known classical fact that the smallest eigenvalue of L_λ^h converges to the smallest eigenvalue of L_λ as $h \rightarrow 0$; cf. [21, 11.2.10]. This implies $f_h \rightarrow f$ pointwise and hence uniformly on any compact set Λ , the latter since f, f_h are concave.

Let us now fix a convex compact Λ with the global maximum $\bar{\lambda}$ of f in its interior. (Notice here that the maximum $\bar{\lambda}$ of f is unique since for any other maximum $\tilde{\lambda}$, the corresponding $\tilde{\mu} = \mu_{\min}(L_{\tilde{\lambda}})$ and $\tilde{v}$ would satisfy the Euler–Lagrange equation (2.4) with boundary condition (2.5), and by Theorem 3.5 would produce an optimal solution for $(\tilde{P})$. This gave $\bar{v} = \tilde{v}$, and hence also $(\bar{\lambda}, \bar{\mu}) = (\tilde{\lambda}, \tilde{\mu})$ by the pseudo-Haar assumption). As $f_h \rightarrow f$ uniformly on Λ , the global maxima λ^h of the f_h , which are unique for h small enough, must belong to Λ for h small enough. Again, this strongly relies on the concavity of f, f_h .

b) Since L_λ is a second-order operator, the smallest eigenvalue $\mu_{\min}(L_\lambda)$ is simple (cf. [12, sect. 6]) and hence so is the smallest eigenvalue $\mu_{\min}(L_\lambda^h)$ for every $h < h_0$ (h_0 depending only on Λ) and all $\lambda \in \Lambda$.

By Proposition 2.2, the eigensolutions are in $W^{1,1+s}(\Omega)$. Using this, another known result tells us that (with a constant C_L independent of h) the smallest eigenvalue $\mu_{\min}(L)$ belonging to a Neumann boundary problem for a V -coercive second-order partial differential operator L satisfies estimates of the form

$$(4.4) \quad |\mu_{\min}(L) - \mu_{\min}(L^h)| \leq C_L h^{2s} \quad \text{and} \quad \|e(L) - e(L^h)\|_V \leq C_L h^s$$

if $e(L), e(L^h)$ are the corresponding normalized eigensolutions; cf. [21, sect. 11]. The difficulty is now that we are dealing with a family L_λ , $\lambda \in \Lambda$, of such operators and will need a uniform estimate of the type (4.4).

Following carefully the arguments in sections 11.2.2 and 11.2.3 of [21], where the estimates (4.4) are obtained for single eigenvalues of general V -coercive elliptic operators, it can in fact be shown that the constant C_{L_λ} for the L_λ is of the form $\mathcal{O}(|\lambda|)$. Therefore, on a compact set Λ , the uniform estimates

$$(4.5) \quad |\mu_{\min}(L_\lambda) - \mu_{\min}(L_\lambda^h)| \leq Ch^{2s} \quad \text{and} \quad \|e(L_\lambda) - e(L_\lambda^h)\|_V \leq Ch^s$$

are satisfied. We do not present the individual steps of the derivation of (4.5) here, since this may be done by tracing the constants in the above reference, while otherwise does not provide any new insight. (Notice, however, that the arguments in [21] are for single eigenvalues, and a similar result for say the finite element approximation of the strongest column problem [13] would require a more sophisticated reasoning.)

Notice that the first formula in (4.5) gives the first part of statement 1. The second part of item 1 now follows readily from the representation (4.1) of the partial derivatives of $\lambda \rightarrow \mu_{\min}(L_\lambda)$, which gives $\partial f(\lambda)/\partial \lambda_i = \int_\Omega a_i e(L_\lambda)^2 dx + b_i$ and $\partial f_h(\lambda)/\partial \lambda_i = \int_\Omega a_i e(L_\lambda^h)^2 dx + b_i$. This proves item 1.

c) The second formula of item 1 implies $|\nabla f(\lambda^h)| = |\nabla f(\lambda^h) - \nabla f_h(\lambda^h)| \leq Ch^s$ for h small enough, and in particular, $\nabla f(\lambda^h) \rightarrow 0$. Now by the continuity of ∇f , every convergent subsequence λ^{h_i} of $\lambda^h \in \Lambda$ must have a limit λ satisfying $\nabla f(\lambda) = 0$. Since the global maximum $\bar{\lambda}$ of f is unique, we deduce that $\lambda^h \rightarrow \bar{\lambda}$.

Now consider any sequence $e^h := e(L_{\lambda^h}^h)$ of normalized eigenfunctions corresponding to $\mu_{\min}(L_{\lambda^h}^h)$. Since the embedding $W^{1,2} \rightarrow \mathcal{L}^2$ is compact, we find a subsequence $e^{h_i} \rightarrow e$ for some $e \in \mathcal{L}^2$. Reasoning exactly as in [21, 11.2.11], we can argue that in fact $e \in W^{1,2}$ and convergence is in the Sobolev norm. In particular, e must be a normalized eigenfunction of $L_{\bar{\lambda}}$, and since there is only one such function, $e(L_{\bar{\lambda}})$, we deduce that the entire sequence converges to this normalized eigenfunction in $W^{1,2}$ -norm. (Again, this part of the reasoning makes strong use of the fact that the smallest eigenvalue is simple.) We deduce that $v^h = b_0^{1/2} e(L_{\lambda^h}^h) \rightarrow \bar{v} = b_0^{1/2} e(L_{\bar{\lambda}})$, which gives the second part of statement 2. The third part of item 2 is clear from $u^h = (v^h)^2$ and $\bar{u} = \bar{v}^2$.

d) We finally obtain the rate of convergence in statement 2. Starting out with

$$\|e(L_{\lambda^h}^h) - e(L_{\bar{\lambda}})\|_V \leq \|e(L_{\lambda^h}^h) - e(L_{\lambda^h})\|_V + \|e(L_{\lambda^h}) - e(L_{\bar{\lambda}})\|_V,$$

we observe that the first term on the right-hand side is $\mathcal{O}(h^s)$ by (4.5). For the second term, observe that $|\lambda^h - \bar{\lambda}| = \mathcal{O}(h^{2s})$, so the claim follows when we observe that $\lambda \rightarrow e_\lambda = e(L_\lambda)$ is Lipschitz in a neighborhood of $\bar{\lambda}$. The latter follows again from the fact that the eigenvalues $\mu_{\min}(L_\lambda)$ are simple, whence the corresponding eigenfunctions $e(L_\lambda)$ depend smoothly on λ ; cf. [25]. This shows $\|v^h - \bar{v}\|_V = b_0^{1/2} \|e(L_{\lambda^h}^h) - e(L_{\bar{\lambda}})\|_V = \mathcal{O}(h^s)$, and hence $\|u^h - \bar{u}\|_{W^{1,1}} = \mathcal{O}(h^s)$. $\square$

Remarks. 1) An analogous result may be derived for the relaxed program (P_r) and its dual (D_r) . The same rate of convergence is obtained, since the argument is based on properties of the function $\mu_{\min}$. We do not present the detailed statement.

2) Similar results for (D) , (D_r) could be obtained for a finite difference approximation. The analogues of (4.4) are now

$$(4.6) \quad |\mu_{\min}(L) - \mu_{\min}(L^h)| \leq C_L h \quad \text{and} \quad \|e(L) - e(L^h)\|_1 \leq C_L h$$

(see [21, sect. 11]), and again one has to show that the constants behave like $C_{L_\lambda} = \mathcal{O}(|\lambda|)$ for the operators L_λ we have in mind. This is once again a problem of tracing constants in a classical derivation of (4.6), as given, e.g., in [21, sect. 11]. (And again, the situation is expected to be more complicated for multiple eigenvalues.)

Let us now consider the discretization of programs (D) , (D_r) . For simplicity we keep the notation $\Delta + \sum_{k=1}^m \lambda_k a_k = \Delta + \sum_{k=1}^m \lambda_k A^k$ for the discretized form of the operator L_λ . Switching from the conventional operator notation (N) to the usual matrix notation, we have $\mu_{\min}(L_\lambda) = -\mu_{\max}(L_\lambda)$, where $\mu_{\max}(L_\lambda)$ now denotes the largest eigenvalue of the matrix L_λ . Passing to the more convenient convex form, the

dual (D) is now a semidefinite program

$$(D) \quad \begin{aligned} & \text{minimize} && \alpha b_0 - \sum_{k=1}^m \lambda_k b_k \\ & \text{subject to} && \alpha I - \Delta - \sum_{k=1}^m \lambda_k A^k \geq 0, \quad \alpha \in \mathbb{R}, \quad \lambda \in \mathbb{R}^m, \end{aligned}$$

which can be solved using the duality theory as presented in [28, Thms. 12.1, 12.2] or [38], references to which we refer the reader for concise information on the subject.

Since the tolerance model (1.4) was based on the Euclidean norm, (D_r) does not directly lead to a semidefinite program. Solving (D_r) as it stands requires techniques from eigenvalue optimization as presented, e.g., in Overton [33] and Overton and Womersley [34].

Let us show how to obtain a semidefinite programming formulation for the restoration problem (1.4). The way to proceed is to replace the tolerance model (P_r) , resp., its dual (D_r) by a penalty-type approach

$$(P_p) \quad \begin{aligned} & \text{maximize} && \mathcal{I}(u) + \frac{C}{2} |Au - b|^2 \\ & \text{subject to} && \int_{\Omega} u \, dx = b_0. \end{aligned}$$

The same duality scheme as applied in previous sections leads to the concave dual

$$(D_p) \quad \text{maximize } h(\lambda) = \mu_{\min}(L_{\lambda}[\cdot])b_0 + \sum_{k=1}^m \lambda_k b_k - \frac{2}{C} |\lambda|^2$$

based on the same return formula (cf. Theorems 3.4, 3.5). The important fact is now that (P_r) and (P_p) are equivalent in the sense that, for a fixed $\epsilon > 0$, the solution $\bar{u}$ of (P_r) with tolerance level ϵ also solves (P_p) for a certain constant $C = C(\epsilon)$, and conversely, if $\hat{u}$ solves (P_p) with a given constant $C > 0$, then it also solves (P_r) with tolerance $\epsilon = \epsilon(C) = |A\hat{u} - b|$. This may be checked by writing down the necessary optimality conditions for both programs.

A disadvantage of $(P_p), (D_p)$ as compared to $(P_r), (D_r)$ is that it is harder to obtain a default value for the constant $C > 0$ than for the tolerance ϵ , the dependence $C = C(\epsilon)$ not being explicit. On the other hand, it is certainly more convenient to deal with (D_p) as far as eigenvalue optimization is concerned. Indeed, we obtain a semidefinite programming formulation for (D_p) by introducing a new variable t together with the constraint $t \geq |\lambda|^2 = \lambda^t \lambda$. The latter translates into $tI \geq \lambda \lambda^t$, which is recognized as a semidefinite constraint (see [38]). Indeed, (D_p) appears as

$$(D_p) \quad \begin{aligned} & \text{minimize} && \alpha b_0 - \sum_{k=1}^m \lambda_k b_k + \frac{2}{C} t \\ & \text{subject to} && \alpha I - \Delta - \sum_{k=1}^m \lambda_k A^k \geq 0, \\ & && \begin{pmatrix} tI & \lambda \\ \lambda^t & 1 \end{pmatrix} \geq 0, \quad \lambda \in \mathbb{R}^m, \quad t, \alpha \in \mathbb{R}. \end{aligned}$$

While the tolerance model had to be replaced by the penalty-type model in order to deal with the Euclidean norm, the original form of (D_r) is suitable if in (1.4)

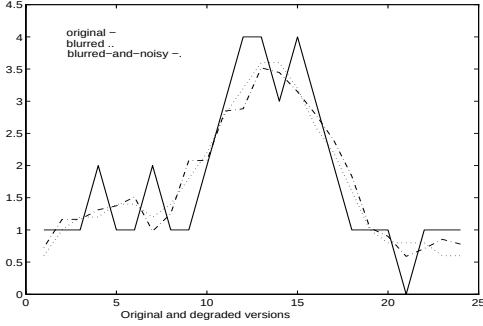


FIG. 1.

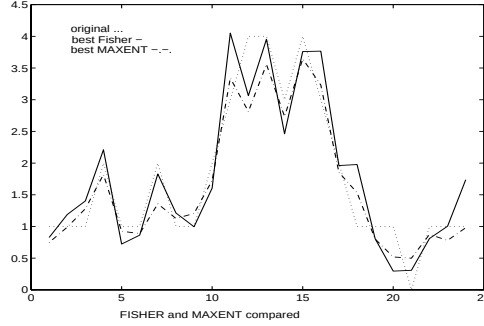


FIG. 2.

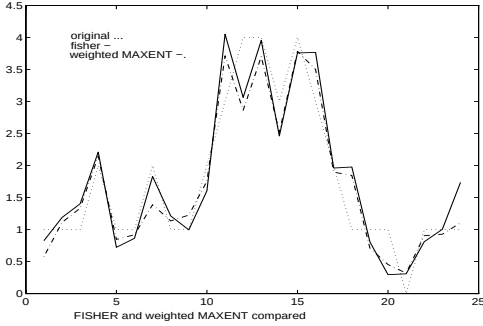


FIG. 3.

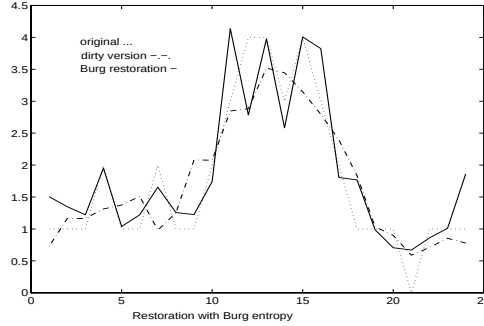


FIG. 4.

polyhedral norms like the one or infinity norm are used. In fact, starting with any norm $|Au - b|_*$ in (1.4) gives the dual norm expression $|\lambda|$ in (D_r) . Now if, for instance, $|\cdot|$ in (D_r) denotes the one norm, a semidefinite programming formulation is obtained by introducing new variables λ_k^+, λ_k^- (satisfying $\lambda_k = \lambda_k^+ - \lambda_k^-$ and $\lambda_k^\pm \geq 0$):

$$\begin{aligned}
 (D_r) \quad & \text{minimize} \quad \alpha b_0 - \sum_{k=1}^m (\lambda_k^+ - \lambda_k^-) b_k + \sum_{k=1}^m (\lambda_k^+ + \lambda_k^-) \\
 & \text{subject to} \quad \alpha I - \Delta - \sum_{k=1}^m (\lambda_k^+ - \lambda_k^-) A^k \geq 0, \\
 & \quad \lambda_k^+ \geq 0, \lambda_k^- \geq 0, \alpha \in \mathbb{R}.
 \end{aligned}$$

Similarly, when $|\cdot|$ in (D_r) denotes the infinity norm, the same decomposition leads to a term $\max_k (\lambda_k^+ + \lambda_k^-)$, which may be transformed to the semidefinite form as for instance in [38, sect. 1].

5. Experiments. This last section presents an experiment where we compared various restoration techniques.

Consider the setting (1.3), where a signal $u(t)$ has been blurred through convolution with a kernel $q(t)$ and by adding a random noise term. As a simulation, the piecewise linear original signal u on $[0, 1]$ displayed in Figure 1 (continuous line) was convolved by a pill-box blur $q(t)$ of variance .5, and white noise of variance $\sigma^2 = .4$ was added, leading to a signal-to-noise ratio of 16dB. The degraded signal v was sampled at 25 equidistant points. The blurred (resp., blurred-and-noisy) images are also shown in Figure 1 as the dotted (resp., broken) line.

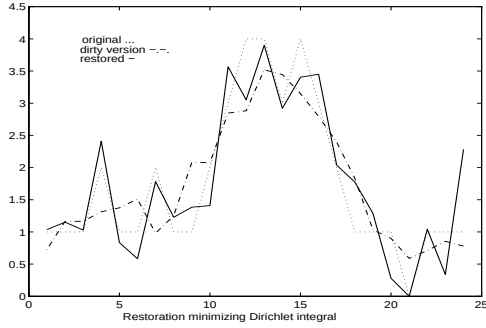


FIG. 5.

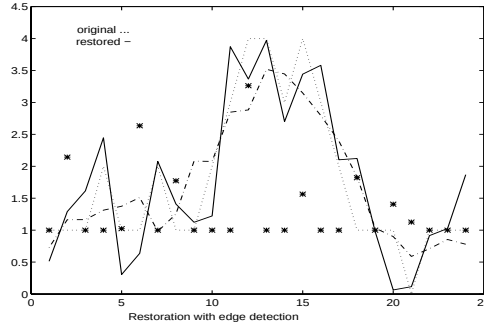


FIG. 6.

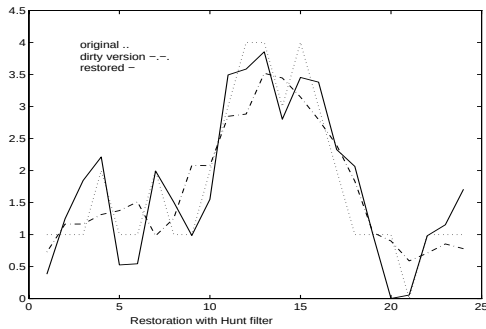


FIG. 7.

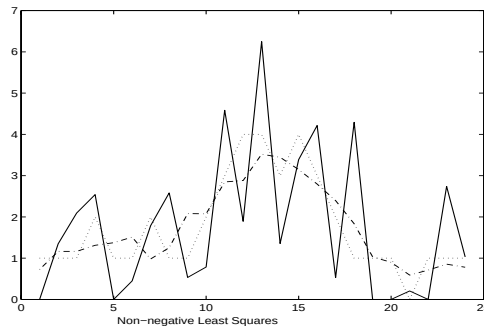


FIG. 8.

The Fisher restoration was obtained on the basis of model (D_p) , using a discretization with $h = 1/50$, and the penalty constant was found experimentally. Figure 2 shows the Fisher restoration (continuous line), compared with a maximum entropy restoration (broken line) based on the model discussed, e.g., in [3, 10, 18, 22] and displayed against the original (dotted line). Figure 3 shows a comparison of the same Fisher restoration with a minimum cross-entropy method, where the dirty signal was used as a priori. As can be seen, with this weighting, the maximum entropy technique can catch up better with the Fisher restoration. For both entropy restorations, the tolerance model (1.4) was used, with the tolerance found experimentally, starting with the default as suggested by Example 1.2.

Figure 4 shows a restoration obtained by Burg's entropy $-\int \log u(x) dx$, displayed against the original and dirty versions. Not unexpectedly, Burg's method produced too spiky results, which agrees with experiments reported, e.g., in [6, 16]. Figure 5 shows the restoration obtained by minimizing the Dirichlet integral $\int |\nabla u(x)|^2 dx$ subject to the constraints (1.4) which for this example works surprisingly well, but generally has a tendency to produce restorations lacking in contrast. Figure 6 shows how this could be improved using a method along the lines of Geman and Geman [19], which combines deconvolution and edge detection. In its 1D discretized form it consists of minimizing a functional of the type

$$f(u, w) = \sum_{k=1}^{n-1} \frac{(u_{k+1} - u_k)^2}{w_k} + \alpha \sum_{k=1}^{n-1} (w_k - 1)$$

subject to the constraints (1.4) and $w_k \geq 1$. Here the variables w_k play the role of switching parameters. The state $w_k = 1$ corresponds to the switch being on, so that the gap $(u_{k+1} - u_k)^2$ is fully penalized. Any value $w_k > 1$ is meant to indicate a structural gap at position k in the unknown image u (or at least a tendency for such), the philosophy being that in this case the penalty for a jump at position k should be gradually released or even switched off. Switching off is paid for through the second term, with the control parameter α ruling the trade-off between the two. Notice that while Geman and Geman [19] use real switches, leading to a discrete variable with states on and off, we prefer a continuous version (using dimmers instead of switches). The states of the variables w_k in Figure 6 are represented by the stars.

Figure 7 shows a restoration obtained by Hunt's filter [23], which is based on minimizing the integral $\int |\Delta u(x)|^2 dx$ and which gives a linear inverse filter if the penalty approach is used. The final Figure 8 was included to show how a direct inversion of (1.3) using least squares badly fails due to the relatively high noise contribution.

Acknowledgments. We thank the editor, Stephen Boyd, and the two anonymous referees for their valuable comments. Helpful discussions with Jean-Baptiste Hiriart-Urruty (Toulouse), Adrian Lewis (Waterloo), and Martin Wursthorn (Stuttgart) during the preparation of this paper are gratefully acknowledged.

REFERENCES

- [1] R. ADAMS, *Sobolev Spaces*, Academic Press, New York, 1975.
- [2] D. L. ANGWIN AND H. KAUFMAN, *Nonhomogeneous image identification and restoration procedures*, Springer Ser. Inform. Sci., 23 (1991), pp. 177–208.
- [3] J. M. BORWEIN AND A. S. LEWIS, *Duality relationships for entropy-like minimization problems*, SIAM J. Control Optim., 29 (1991), pp. 325–338.
- [4] J. M. BORWEIN AND A. S. LEWIS, *Partially finite convex programming I, II*, Math. Programming, 57 (1992), pp. 15–48 and 49–84.
- [5] J. M. BORWEIN, A. S. LEWIS, AND D. NOLL, *Maximum entropy reconstruction using derivative information I: Fisher information and convex duality*, Math. Oper. Res., 21 (1996), pp. 442–468.
- [6] J. M. BORWEIN, A. S. LEWIS, M. N. LIMBER, AND D. NOLL, *Maximum entropy reconstruction using derivative information II: Computational results*, Numer. Math., 69 (1995), pp. 243–256.
- [7] J. M. BORWEIN, M. N. LIMBER, AND D. NOLL, *Fast heuristic methods for function reconstruction using derivative information*, Appl. Anal., 58 (1995), pp. 241–261.
- [8] P. J. BROCKWELL AND R. A. DAVIS, *Time Series: Theory and Methods*, 2nd ed., Springer-Verlag, New York, 1991.
- [9] K. J. BROWN AND S. S. LIN, *On the existence of positive eigenfunctions for an eigenvalue problem with indefinite weight functions*, J. Math. Anal. Appl., 75 (1980), pp. 112–120.
- [10] S. F. BURCH, S. F. GULL, AND J. K. SKILLING, *Image restoration by a powerful maximum entropy method*, Computer Vision, Graphics, and Image Processing, 23 (1983), pp. 113–128.
- [11] P. G. CIARLET, *The Finite Element Method for Elliptic Problems*, North-Holland, Amsterdam, 1978.
- [12] R. COURANT, *Methoden der mathematischen Physik I*, Springer Verlag, Berlin, 1931.
- [13] S. J. COX AND M. L. OVERTON, *Optimal design of a column against buckling*, SIAM J. Math. Anal., 23 (1992), pp. 287–325.
- [14] B. DACOROGNA, *Direct Methods in the Calculus of Variations*, Springer-Verlag, Berlin, 1989.
- [15] J. L. DOOB, *Probability and statistics*, Trans. Amer. Math. Soc., 36 (1934), pp. 759–775.
- [16] G. J. ERICKSON AND C. R. SMITH, EDS., *Maximum-Entropy and Bayesian Methods in Science and Engineering*, Kluwer, Dordrecht, The Netherlands, 1988.
- [17] R. A. FISHER, *Theory of statistical estimation*, Philos. Trans. Roy. Soc. London, 222 (1930), pp. 309–368.
- [18] B. R. FRIEDEN, *Restoring with maximum likelihood and maximum entropy*, J. Opt. Soc. Amer., G2 (1972), pp. 511–518.

- [19] S. GEMAN AND D. GEMAN, *Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images*, IEEE Trans. Pattern Analysis and Machine Intelligence, 6 (1984), pp. 721–741.
- [20] B. GINSBURG AND A. IOFFE, *The maximum principle in optimal control of systems governed by semilinear equations*, in Proc. IMA Workshop on Nonsmooth Analysis and Geometric Methods in Deterministic Optimal Control, B. Mordukhovich and B. Sussman, eds., Springer-Verlag, 1995.
- [21] W. HACKBUSCH, *Elliptic Differential Equations: Theory and Numerical Treatment*, Springer-Verlag, New York, 1992.
- [22] R. M. HARALICK, E. ØSTEVOLD, AND X. ZHUANG, *A differential equation approach to maximum entropy image reconstruction*, IEEE Trans. Acoust. Speech Signal Process., 35 (1987), pp. 208–218.
- [23] B. R. HUNT, *Deconvolution of linear systems by constrained regression and its relationship to the Wiener theory*, IEEE Trans. Automat. Control, 23 (1972), pp. 703–705.
- [24] R. W. JOHNSON AND J. E. SHORE, *Which is the better entropy expression for speech processing: $-s \log s$ or $\log s$?*, IEEE Trans. Acoust., Speech Signal Proc., 32 (1984), pp. 129–137.
- [25] T. KATO, *Perturbation Theory for Linear Operators*, 2nd ed., Springer-Verlag, New York, 1984.
- [26] A. K. KATSAGGELOS AND K.-T. LAY, *Maximum likelihood identification and restoration of images using the expectation-maximization algorithm*, Springer Series in Information Sciences, 23 (1991), pp. 143–176.
- [27] A. S. LEWIS, *The convergence of entropy-based approximations for moment problems*, Optimization, 28 (1994), pp. 383–395.
- [28] A. S. LEWIS AND M. L. OVERTON, *Eigenvalue optimisation*, Acta Numerica, (1996).
- [29] M. K. V. MURTHY AND G. STAMPACCHIA, *A variational inequality with mixed boundary conditions*, Israel J. Math., 13 (1972), pp. 188–224.
- [30] D. NOLL, *Rates of convergence for best entropy estimates*, Stat. Decisions, 13 (1995), pp. 141–165.
- [31] D. NOLL, *Restoration of degraded images with maximum entropy*, J. Global Optim., 10 (1997), pp. 91–103.
- [32] D. NOLL, *Consistency of a nonlinear deconvolution method with applications in image restoration*, Adv. Math. Sci. Appl., 7 (1997), pp. 789–808.
- [33] M. L. OVERTON, *Large scale optimisation of eigenvalues*, SIAM J. Optim., 2 (1992), pp. 88–120.
- [34] M. L. OVERTON AND R. S. WOMERSLEY, *Second derivatives for optimizing eigenvalues of symmetric matrices*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 697–718.
- [35] R. T. ROCKAFELLAR, *Integrals which are convex functionals I, II*, Pacific J. Math., 24 and 39 (1968), pp. 525–539 and 439–469.
- [36] R. N. SILVER, *Quantum Statistical Inference*, IEEE Report, Kluwer, Dordrecht, The Netherlands, 1992.
- [37] H. J. TRUSSELL, *The relationship between image restoration by the maximum a posteriori method and a maximum entropy method*, IEEE Trans. Acous. Speech Signal Proc., 28 (1980), pp. 114–117.
- [38] L. VANDENBERGHE AND S. BOYD, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.
- [39] W. P. ZIEMER, *Weakly Differentiable Functions*, Springer-Verlag, New York, 1989.
- [40] V. ZIVOJNOVIC, *Higher-order statistics and Huber’s robustness*, in IEEE Signal Processing Workshop on Higher-Order Statistics, Spain, 1995.

ON SOME PROPERTIES OF QUADRATIC PROGRAMS WITH A CONVEX QUADRATIC CONSTRAINT*

STEFANO LUCIDI[†], LAURA PALAGI[†], AND MASSIMO ROMA[†]

Abstract. In this paper we consider the problem of minimizing a (possibly nonconvex) quadratic function with a quadratic constraint. We point out some new properties of the problem. In particular, in the first part of the paper, we show that (i) given a KKT point that is not a global minimizer, it is easy to find a “better” feasible point; (ii) strict complementarity holds at the local-nonglobal minimizer. In the second part of this paper, we show that the original constrained problem is equivalent to the unconstrained minimization of a piecewise quartic merit function. Using the unconstrained formulation we give, in the nonconvex case, a new second order necessary condition for global minimizers. In the third part of this paper, algorithmic applications of the preceding results are briefly outlined and some preliminary numerical experiments are reported.

Key words. quadratic function, ℓ_2 -norm constraint, merit function

AMS subject classifications. 90C30, 65K05

PII. S1052623494278049

1. Introduction. In this paper we study the problem of minimizing a general quadratic function $q : \mathbb{R}^n \rightarrow \mathbb{R}$ subject to an ellipsoidal constraint, that is,

$$(1) \quad \min\{q(x) : x^T H x \leq a^2\},$$

where H is a symmetric positive definite $n \times n$ matrix and a is a positive scalar. The interest in this problem initially arose in the context of trust region methods for solving unconstrained optimization problems. In fact, such methods require at each iteration an approximate solution of problem (1) where $q(x)$ is a local quadratic model of the objective function over a restricted ellipsoidal region centered around the current iterate. However, recently, it has been shown that problems with the same structure of (1) play an important role not only in the field of unconstrained minimization. In fact, the solution of problem (1) is at the basis of algorithms for solving general constrained nonlinear problems (e.g., [3, 20, 27, 42]) and integer programming problems (e.g., [19, 21, 22, 31, 41]). Many papers have been devoted to pointing out the specific features of problem (1). Among the most important results there are the necessary and sufficient conditions for a point x to be a global minimizer, due to Gay [12] and Sorensen [35], and the characterization and uniqueness of the local-nonglobal minimizer, due to Martínez [26]. The particular structure of problem (1) has led to the development of algorithms for finding a global solution. The first algorithms proposed in literature were those of Gay and Sorensen [12, 35]. Moré and Sorensen [30] developed an algorithm that produces an approximate global minimizer in a finite number of steps. More recently, it has been proved that an approximation to the global solution can be computed in polynomial time (see, for example, [21, 38, 39, 40, 41]). Furthermore, Moré [28] has considered a more general case by allowing in problem (1) a general quadratic constraint and has extended the results of [12, 30, 35].

In spite of all these results, there is still interest in studying problem (1). In fact, as we mentioned before, there is a growing use for problem (1) as a tool for tackling large

* Received by the editors November 30, 1994; accepted for publication (in revised form) December 2, 1996. This work was partially supported by Agenzia Spaziale Italiana, Roma, Italy.
<http://www.siam.org/journals/siopt/8-1/27804.html>

[†] Università di Roma “La Sapienza,” Dipartimento di Informatica e Sistemistica, Via Buonarroti 12, I-00185 Roma, Italy (lucidi@dis.uniroma1.it, palagi@dis.uniroma1.it, roma@dis.uniroma1.it).

nonlinear programming problems and combinatorial optimization problems. This leads to the necessity of solving more and more efficiently large scale problems of the type (1) and motivates further research on theoretical properties of problem (1) and on the definition of efficient methods for locating its global minimizers. Recently some interesting algorithms for tackling large scale trust region problems have been proposed in [33, 34, 36]. The basic idea behind these algorithms is to recast the trust region problem in terms of a parametrized eigenvalue problem and then to adjust the parameter to find an optimal solution.

In this paper we point out further theoretical properties of problem (1). In particular, our research develops along two lines: the study of some new properties of its KKT points and its equivalence to an unconstrained minimization problem. Besides their own theoretical interest, these results allow us to define new classes of algorithms for solving large scale case trust region problems. These algorithms use only matrix–vector product and do not require the solution of an eigenvalue problem at each iteration (see [24] for details). The paper is organized as follows. In section 2 we recall some preliminary results. In section 3 we show that

(i) given a KKT point $\bar{x}$, which is not a global minimizer, it is possible to find a new feasible point $\hat{x}$ such that the objective function is strictly decreased, i.e., $q(\hat{x}) < q(\bar{x})$;

(ii) the strict complementarity condition holds at the local minimizer, hence, in the nonconvex case, strict complementarity holds at local and global minimizers.

In section 4 we show that there is a one to one correspondence between KKT points (global minimizers) of problem (1) and stationary points (global minimizers) of a piecewise quartic merit function $P : \mathbb{R}^n \rightarrow \mathbb{R}$. Therefore, problem (1) is equivalent to the unconstrained problem of P over $\mathbb{R}^n$. In section 5, by exploiting some results of the preceding sections, we give a new second order necessary condition for global minimizers of problem (1). Finally, in section 6, we sketch some possible applications of the results of section 3 and section 4 for defining new classes of algorithms for solving large scale trust region problems.

In the rest of this paper we will use the following notation. Given a vector $x \in \mathbb{R}^n$, we denote by $\|x\|$ the ℓ_2 -norm on $\mathbb{R}^n$. The ℓ_2 -norm of a $n \times n$ matrix Q is defined by $\|Q\| = \sup\{\|Qx\| : \|x\| = 1\}$. Moreover, we denote by $\mu_1 \leq \mu_2 \leq \dots \leq \mu_n$ the eigenvalues of Q .

2. Preliminaries. Without loss of generality we can assume that the feasible set $\mathcal{F}$ is defined by

$$\mathcal{F} = \left\{ x \in \mathbb{R}^n : \|x\|^2 \leq a^2 \right\},$$

so that the problem under consideration is

$$(2) \quad \min\{q(x) : \|x\|^2 \leq a^2\},$$

where $q : \mathbb{R}^n \rightarrow \mathbb{R}$ is given by

$$(3) \quad q(x) = \frac{1}{2}x^T Qx + c^T x$$

and Q is an $n \times n$ symmetric matrix and $c \in \mathbb{R}^n$.

In fact, since H is positive definite, we can reduce problem (1) to the form (2) by employing the transformation $y = H^{\frac{1}{2}}x$ (however, we refer to [25] for the direct

treatment of problem (1)). The Lagrangian function associated with problem (2) is the function

$$L(x, \lambda) = \frac{1}{2}x^T Qx + c^T x + \lambda(\|x\|^2 - a^2).$$

A KKT point for problem (2) is a pair $(\bar{x}, \bar{\lambda}) \in \mathbb{R}^n \times \mathbb{R}$ such that

$$(4) \quad \begin{aligned} (Q + 2\bar{\lambda}I)\bar{x} &= -c, \\ \bar{\lambda}(\|\bar{x}\|^2 - a^2) &= 0, \\ \bar{\lambda} &\geq 0, \quad \|\bar{x}\|^2 \leq a^2. \end{aligned}$$

Furthermore, we say that strict complementarity holds at a KKT pair $(\bar{x}, \bar{\lambda})$ if $\bar{\lambda} > 0$ for $\|\bar{x}\|^2 = a^2$.

It is well known that it is possible to completely characterize the global solutions of problem (2) without requiring any convexity assumption on the objective function. In fact, the following result due to Gay [12] and Sorensen [35] holds (see also Vavasis [38]).

PROPOSITION 2.1. *A point x^* such that $\|x^*\|^2 \leq a^2$ is a global solution of problem (2) if and only if there exists a unique $\lambda^* \geq 0$ such that the pair (x^*, λ^*) satisfies the KKT conditions*

$$\begin{aligned} (Q + 2\lambda^*I)x^* &= -c, \\ \lambda^*(\|x^*\|^2 - a^2) &= 0 \end{aligned}$$

and the matrix $(Q + 2\lambda^*I)$ is positive semidefinite. If $(Q + 2\lambda^*I)$ is positive definite, then problem (2) has a unique global solution.

Moreover, Martínez [26] gave the following characterization of the local-nonglobal minimizer for problem (2).

PROPOSITION 2.2. *There exists at most one local-nonglobal minimizer $\bar{x}$ of problem (2). Moreover, we have $\|\bar{x}\| = a^2$ and the KKT necessary conditions hold with $\bar{\lambda} \in (-\mu_2, -\mu_1)$, where $\mu_1 < \mu_2$ are the two smallest eigenvalues of Q .*

3. Further features of KKT points. In this section we give some new properties of the KKT points for problem (2). Our interest in the characterization of KKT points is due to the fact that, in general, algorithms for the solution of constrained problems converge toward KKT points. We show that the number of different values that the objective function can take at KKT points is bounded from above by the number of negative eigenvalues of the matrix Q . First we state a preliminary result that extends one given in [38].

LEMMA 3.1. *Let $(\hat{x}, \lambda)$ and $(\bar{x}, \lambda)$ be KKT pairs for problem (2) with the same KKT multiplier. Then $q(\hat{x}) = q(\bar{x})$.*

Proof. We observe that the function $q(x)$ can be rewritten at every KKT pair (x, λ) as follows:

$$q(x) = \frac{1}{2}c^T x - \lambda\|x\|^2.$$

By using the KKT conditions we obtain

$$q(\hat{x}) = \frac{1}{2}c^T \hat{x} - \lambda\|\hat{x}\|^2 = -\frac{1}{2}\bar{x}^T (Q + 2\lambda I) \hat{x} - \lambda a^2 = \frac{1}{2}c^T \bar{x} - \lambda\|\bar{x}\|^2 = q(\bar{x}). \quad \square$$

Now we can state the following proposition whose proof follows from a result of Forsythe and Golub [11] on the number of stationary values of a second-degree polynomial on the unit sphere. For sake of completeness we give a sketch of the proof.

PROPOSITION 3.2. *There exist at most $\min\{2m + 2, 2n + 1\}$ KKT points with distinct multipliers λ , where m is the number of negative distinct eigenvalues of Q .*

Proof. First we observe that at every KKT point (x, λ) such that $\|x\|^2 < a^2$ the value of the objective function q is constant. This easily follows from Lemma 3.1 by observing that all these pairs are characterized by the fact that $\lambda = 0$. Now we consider the values of the function $q(x)$ at all the points such that $\|x\|^2 = a^2$. Since Q is symmetric, there exists an orthogonal matrix V such that $V^T Q V = \text{diag}_{i=1, \dots, n} \{\mu_i\}$, where $\mu_1 \leq \mu_2 \leq \dots \leq \mu_n$ are the eigenvalues of Q . By considering the transformation $x = V\alpha$ we can write the first equation of the KKT condition (4) (premultiplied by V^T) as follows:

$$(5) \quad (\text{diag}_{i=1, \dots, n} \{\mu_i\} + 2\lambda I) \alpha = -\beta$$

with $\beta = V^T c$. Hence, recalling that $\|x\|^2 = \|V\alpha\|^2 = a^2$, we have that the KKT multipliers must satisfy the system

$$(6) \quad \begin{aligned} g(\lambda) &= a^2, \\ \lambda &\geq 0, \end{aligned}$$

where

$$(7) \quad g(\lambda) = \sum_{i=1}^n \frac{\beta_i^2}{(\mu_i + 2\lambda)^2}.$$

The function $g(\lambda)$ has poles at $-1/2\mu_i$ and it is convex on the subintervals $[-1/2\mu_{i-1}, -1/2\mu_i]$. Thus there exist at most 2 roots of $g(\lambda) = a^2$ in each subinterval. Moreover, since $g(\lambda) \rightarrow 0$ as $\lambda \rightarrow \pm\infty$, the equation $g(\lambda) = a^2$ has one root in each extreme subinterval. If all eigenvalues μ_i are positive, there exists at most one nonnegative root; if all the eigenvalues are negative, there are at most $2n$ nonnegative roots; in the case of $m < n$ negative eigenvalues, there are at most $2m + 1$ nonnegative roots. Hence the number of the solutions of system (6) is at most $\min\{2m + 1, 2n\}$.

Finally, by summarizing the two cases, we can conclude that the number of distinct KKT multipliers is bounded above by $\min\{2m + 1, 2n\} + 1$. $\square$

Recalling Lemma 3.1, we get the following corollary directly.

COROLLARY 3.3. *The number of distinct values of the objective function $q(x)$ at KKT points is bounded from above by $\min\{2m + 2, 2n + 1\}$.*

Now we can state the main result of this section. In particular, we show that the peculiarity of problem (2) can be exploited to escape from the KKT points that are not global solutions in the sense that, whenever we have a KKT point $\bar{x}$, either $\bar{x}$ is a global minimizer of problem (2) or it is possible to compute the expression of a feasible point with a strictly lower value of the objective function. This result is very appealing from a computational point of view, as discussed in section 6.

PROPOSITION 3.4. *Let $(\bar{x}, \bar{\lambda})$ be a KKT point for problem (2). Let us define the point $\hat{x}$ in the following way.*

(a) *If $c^T \bar{x} > 0$, then*

$$\hat{x} = -\bar{x}.$$

- (b) If $c^T \bar{x} \leq 0$ and a vector $z \in \mathbb{R}^n$ such that $z^T(Q + 2\bar{\lambda}I)z < 0$ exists, then
 (i) if $\|\bar{x}\|^2 < a^2$,

$$\hat{x} = \bar{x} + \alpha z$$

with

$$0 < \alpha \leq \frac{-z^T \bar{x} + \left[(z^T \bar{x})^2 + (a^2 - \|\bar{x}\|^2) \|z\|^2 \right]^{\frac{1}{2}}}{\|z\|^2};$$

- (ii) if $\|\bar{x}\|^2 = a^2$ and $\bar{x}^T z \neq 0$,

$$\hat{x} = \bar{x} - 2 \frac{\bar{x}^T z}{\|z\|^2} z;$$

- (iii) if $\|\bar{x}\|^2 = a^2$ and $\bar{x}^T z = 0$,

$$\hat{x} = \bar{x} - 2 \frac{a^2}{a^2 + \alpha^2 \|z\|^2} (\bar{x} + \alpha z)$$

with

$$\alpha > \frac{c^T z - \left[(c^T z)^2 + (c^T \bar{x})(z^T(Q + 2\bar{\lambda}I)z) \right]^{\frac{1}{2}}}{z^T(Q + 2\bar{\lambda}I)z}.$$

Then we have $q(\hat{x}) < q(\bar{x})$ and $\|\hat{x}\|^2 \leq a^2$.

Proof. In case (a), the point $\hat{x}$ is still feasible and

$$q(\hat{x}) = \frac{1}{2} \hat{x}^T Q \hat{x} + c^T \hat{x} = \frac{1}{2} \bar{x}^T Q \bar{x} - |c^T \bar{x}| = q(\bar{x}) - 2|c^T \bar{x}| < q(\bar{x}).$$

Now consider case (b). In case (i) we have by the KKT conditions that $\bar{\lambda} = 0$ and hence we have that z is a vector of negative curvature for $q(x)$. Therefore, for every $\alpha > 0$, the point $\hat{x} = \bar{x} + \alpha z$ satisfies the inequality

$$q(\bar{x} + \alpha z) = q(\bar{x}) + \frac{1}{2} \alpha^2 z^T Q z < q(\bar{x}).$$

In particular, if we take $\alpha \leq \tilde{\alpha}$ with

$$\tilde{\alpha} = \frac{-z^T \bar{x} + \left[(z^T \bar{x})^2 + \|\bar{x}\|^2 - a^2 \|z\|^2 \right]^{\frac{1}{2}}}{\|z\|^2},$$

we have that $\|\hat{x}\|^2 \leq a^2$.

Now let us consider case (ii). Let $\hat{x}$ be the vector defined as

$$\hat{x} = \bar{x} - 2 \frac{\bar{x}^T z}{\|z\|^2} z,$$

and consider the quadratic function

$$(8) \quad \hat{L}(x, \bar{\lambda}) = \frac{1}{2} x^T (Q + 2\bar{\lambda}I) x + c^T x.$$

We note that $\|\hat{x}\|^2 = a^2$ and that z is a negative curvature direction for the quadratic function $\hat{L}(x, \bar{\lambda})$. By simple calculation, taking into account that $(Q + 2\bar{\lambda}I)\bar{x} = -c$, we get

$$\hat{L}(\hat{x}, \bar{\lambda}) = \hat{L}(\bar{x}, \bar{\lambda}) + 2 \frac{|\bar{x}^T z|^2}{\|z\|^2} z^T (Q + 2\bar{\lambda}I) z,$$

and hence $\hat{L}(\hat{x}, \bar{\lambda}) < \hat{L}(\bar{x}, \bar{\lambda})$. Hence, recalling the expression (8), we can write

$$q(\hat{x}) < q(\bar{x}) + \frac{1}{2} \bar{\lambda} (\|\bar{x}\|^2 - \|\hat{x}\|^2) = q(\bar{x}).$$

Hence we get the result for case (ii).

Let us consider the case (iii). Let us define the vector $\bar{s} = \bar{x} + \alpha z$ with $\alpha > 0$. We can find a value for α such that $\bar{s}$ is a negative curvature direction for $\hat{L}(x, \bar{\lambda})$ and $\bar{s}^T \bar{x} \neq 0$ so that we can proceed as in case (ii). In fact, by simple calculation we have

$$\bar{s}^T \bar{x} = (\bar{x} + \alpha z)^T \bar{x} = \bar{x}^T \bar{x} = a^2,$$

and by using the KKT conditions

$$\begin{aligned} \bar{s}^T (Q + 2\bar{\lambda}I) \bar{s} &= \bar{x}^T (Q + 2\bar{\lambda}I) \bar{x} + \alpha^2 z^T (Q + 2\bar{\lambda}I) z + 2\alpha \bar{x}^T (Q + 2\bar{\lambda}I) z \\ &= -\alpha^2 |z^T (Q + 2\bar{\lambda}I) z| - 2\alpha c^T z + |c^T \bar{x}|. \end{aligned}$$

By solving the quadratic equation with respect to α we get that $\bar{s}^T (Q + 2\bar{\lambda}I) \bar{s} < 0$ for all $\alpha > \bar{\alpha}$, where

$$\bar{\alpha} = \frac{-c^T z + [(c^T z)^2 + |c^T \bar{x}| |z^T (Q + 2\bar{\lambda}I) z|]^{\frac{1}{2}}}{|z^T (Q + 2\bar{\lambda}I) z|}.$$

Hence, by proceeding as in case (ii), we get the result by introducing the point

$$\hat{x} = \bar{x} - 2 \frac{\bar{x}^T (\bar{x} + \alpha z)}{(\bar{x} + \alpha z)^T (\bar{x} + \alpha z)} (\bar{x} + \alpha z)$$

with $\alpha > \bar{\alpha}$. $\square$

Remark. We note that the local-nonglobal minimizer can correspond either to the case (a) with $\|\bar{x}\|^2 = a^2$ or to the case (b)(ii).

The preceding proposition shows that, if the KKT point $\bar{x}$ is not a global minimizer, it is possible to determine a feasible point $\hat{x}$ such that $q(\hat{x}) < q(\bar{x})$ by computing at most a direction z such that $z^T (Q + 2\bar{\lambda}I) z < 0$. The existence of such a direction is guaranteed by Proposition 2.1 and, from the numerical point of view, its computation is not an expensive task. In fact, we can obtain such a direction by using, for example, the Bunch–Parlett decomposition [2, 29], modified Cholesky factorizations [10], or, for large scale problems, methods based on Lanczos algorithms [4].

Now, as the last result of this section, we investigate a regularity property of the local and global minimizers. In particular, we focus our attention on the strict complementarity property that, roughly speaking, indicates that these points are “really constrained.” Also, this property can be interesting from an algorithmic point of view.

PROPOSITION 3.5. *At the local-nonglobal minimizer for problem (2) the strict complementarity condition holds.*

Proof. Since $\bar{x}$ is a local minimizer, the KKT conditions (4) hold. Moreover the second order necessary conditions require that

$$(9) \quad z^T(Q + 2\bar{\lambda}I)z \geq 0 \quad \text{for all } z : \quad z^T\bar{x} = 0.$$

By Proposition 2.2 we have that $\bar{\lambda} \in (-\mu_2, -\mu_1)$ and $\|\bar{x}\|^2 = a^2$. Obviously if $\mu_1, \mu_2 \geq 0$ there is no local-nonglobal minimizer. Furthermore, if $\mu_1 < 0$ and $\mu_2 \leq 0$, necessarily $\bar{\lambda} > 0$. So we can restrict ourselves to the case

$$\mu_1 < 0, \quad \mu_2 > 0$$

since in this case $0 \in (-\mu_2, -\mu_1)$. Let us assume by contradiction that $\bar{\lambda} = 0$. From (4) and Proposition 2.2 we have that

$$\begin{aligned} \nabla q(\bar{x}) &= 0, \\ z^T Q z &\geq 0 \quad \text{for all } z : \quad z^T \bar{x} = 0, \\ \bar{\lambda} &= 0, \quad \|\bar{x}\|^2 = a^2. \end{aligned}$$

Since $\bar{x}$ is not a global minimizer, by Proposition 2.1 there exists a direction y such that $y^T Q y < 0$ and, from the second order necessary conditions, $y^T \bar{x} \neq 0$. We assume, without loss of generality, that $y^T \bar{x} < 0$. Let us consider the point $x(\alpha) = \bar{x} + \alpha y$ with $\alpha > 0$. We prove that for sufficiently small values of α the point $x(\alpha)$ is feasible and produces a smaller value of the objective function, thus contradicting the assumption of local optimality. In fact, we have

$$\|x(\alpha)\|^2 = \|\bar{x}\|^2 + 2\alpha y^T \bar{x} + \alpha^2 \|y\|^2$$

and hence for $\alpha < 2|y^T \bar{x}|/\|y\|^2$ we obtain $\|x(\alpha)\|^2 < a^2$. Moreover,

$$q(x(\alpha)) = q(\bar{x}) + \alpha \nabla q(\bar{x})^T y + \frac{1}{2} \alpha^2 y^T Q y = q(\bar{x}) + \frac{1}{2} \alpha^2 y^T Q y < q(\bar{x}). \quad \square$$

By this proposition and by Proposition 2.1 we directly obtain the following result.

PROPOSITION 3.6. *In the nonconvex case at every local or global minimizer the strict complementarity holds.*

4. Unconstrained formulation. In this section, we show that problem (2) is equivalent to an unconstrained minimization problem of a piecewise quartic merit function. A general constrained optimization problem can be transformed into an unconstrained problem by defining a continuously differentiable exact penalty function by following, for example, the approach proposed in [6, 7]. However, in the special case of minimization of a quadratic function with box constraints, it has been shown in [16] and [23] that it is possible to define simpler penalty functions by exploiting the particular structure of the problem. In the same spirit of these papers we show that also for problem (2) it is possible to construct a particular continuously differentiable penalty function. This new penalty function takes full advantage of the peculiarities of the trust region problem and enjoys distinguishing features that make its unconstrained minimization significantly simpler in comparison with the unconstrained minimization of the penalty functions proposed in [6, 7]. The main properties of the penalty function proposed in this section are the following.

- It is *globally exact* according to the definition of [7];
- it does not require any shifted barrier term, hence it is defined on the whole space;

- it has a very simple expression (it is piecewise quartic);
- it is known, a priori, for which values of the penalty parameter the correspondence between the constrained problem and the unconstrained one holds.

As a first step for the definition of the exact penalty function, we recall the Hestenes–Powell–Rockafellar augmented Lagrangian function [18, 32]

$$L_a(x, \lambda; \varepsilon) = q(x) + \frac{\varepsilon}{4} \left[\max \left\{ 0, \frac{2}{\varepsilon} (\|x\|^2 - a^2) + \lambda \right\}^2 - \lambda^2 \right],$$

where $\lambda \in \mathbb{R}^m$ and ε is a given positive parameter. Now, according to the classical approach, we replace the multiplier vector λ in the function $L_a(x, \lambda; \varepsilon)$ with a multiplier function $\lambda(x) : \mathbb{R}^n \rightarrow \mathbb{R}$, which yields an estimate of the multiplier vector associated with problem (2) as a function of the variables x . In the literature, different multiplier functions have been proposed (see, e.g., [6, 7, 9, 13, 23]). However, all the expressions of the multiplier functions given in [6, 7, 9, 13] are not defined in the origin of the space. Here we define a new simpler multiplier function that is defined on the whole space $\mathbb{R}^n$ whose expression is the following

$$(10) \quad \lambda(x) = -\frac{1}{2a^2} (x^T Q x + c^T x).$$

Its properties are summarized in the following proposition.

PROPOSITION 4.1.

- (i) $\lambda(x)$ is continuously differentiable with gradient

$$\nabla \lambda(x) = -\frac{1}{2a^2} (2Qx + c).$$

- (ii) If $(\bar{x}, \bar{\lambda})$ is a KKT point for problem (2), then we have $\lambda(\bar{x}) = \bar{\lambda}$.

- (iii) For every $x \in \mathbb{R}^n$ we have $x^T \nabla q(x) + 2\lambda(x) \|x\|^2 = 2\lambda(x)(\|x\|^2 - a^2)$.

Proof. Part (i) easily follows from the definition of the multiplier function (10).

As regards part (ii), from (4) we have that a pair $(\bar{x}, \bar{\lambda})$ satisfies

$$(11) \quad \bar{x}^T Q \bar{x} + c^T \bar{x} + 2\bar{\lambda} \|\bar{x}\|^2 = 0.$$

It is easy to see that if $\|\bar{x}\|^2 = a^2$, (11) corresponds exactly to the definition of the multiplier function (10). Otherwise, if $\|\bar{x}\|^2 < a^2$, (4) imply that $\bar{\lambda} = 0$ and, hence, by comparing (11) and (10) that $\lambda(\bar{x}) = 0$. By simple calculations we have

$$\begin{aligned} x^T \nabla q(x) + 2\lambda(x) \|x\|^2 &= x^T Q x + c^T x - \frac{\|x\|^2}{a^2} (x^T Q x + c^T x) \\ &= -\frac{1}{2a^2} (x^T Q x + c^T x) 2(\|x\|^2 - a^2) = 2\lambda(x)(\|x\|^2 - a^2). \quad \square \end{aligned}$$

On the basis of the previous considerations we can replace the vector λ in the function L_a with the multiplier function $\lambda(x)$. Furthermore, as regards the penalty parameter ε , we can select, a priori, an interval of suitable values depending on the problem data Q, c, a . Therefore, we are now ready to define our merit function $P(x) = L_a(x, \lambda(x); \varepsilon(Q, c, a))$; that is,

$$(12) \quad P(x) = q(x) + \frac{\varepsilon}{4} \left[\max \left\{ 0, \frac{2}{\varepsilon} (\|x\|^2 - a^2) + \lambda(x) \right\}^2 - \lambda^2(x) \right],$$

where $\lambda(x)$ is the quadratic function given by (10) and ε is any parameter that satisfies the following inequality:

$$(13) \quad 0 < \varepsilon < \frac{16a^4}{a^2(8\|Q\| + 3) + 5\|c\|^2}.$$

First, we show some immediate properties of the merit function P .

PROPOSITION 4.2.

(i) $P(x)$ is continuously differentiable with gradient

$$\nabla P(x) = Qx + c - \frac{\varepsilon}{2}\lambda(x)\nabla\lambda(x) + \frac{\varepsilon}{2}\max\left\{0, \frac{2}{\varepsilon}(\|x\|^2 - a^2) + \lambda(x)\right\}\left(\frac{4}{\varepsilon}x + \nabla\lambda(x)\right).$$

(ii) $P(x)$ is twice continuously differentiable except at points where

$$\frac{2}{\varepsilon}(\|x\|^2 - a^2) + \lambda(x) = 0.$$

(iii) $P(x)$ is twice continuously differentiable in a neighborhood of a KKT point $\bar{x}$ where strict complementarity holds.

(iv) For every x such that $\|x\|^2 \leq a^2$ we have that $P(x) \leq q(x)$.

(v) The penalty function $P(x)$ is coercive, and hence it admits a global minimizer.

Proof. Parts (i), (ii), and (iii) directly follow from the expression of the penalty function P . As regards Part (iv), it follows from a classical result on penalty functions (see Theorem 2 of [7]). As regards part (v), we want to show that as $\|x\| \rightarrow \infty$ the function $P(x)$ goes to infinity. First, we observe that

$$\frac{2}{\varepsilon}(\|x\|^2 - a^2) + \lambda(x) \geq \left(\frac{2}{\varepsilon} - \frac{1}{2a^2}\|Q\|\right)\|x\|^2 - \frac{1}{2}\|c\|\|x\| - \frac{2a^2}{\varepsilon},$$

hence, for sufficiently large values of $\|x\|$, the leading term of the preceding inequality is strictly positive since, recalling that ε satisfies (13), we have that $\varepsilon \leq 4a^2/\|Q\|$. Then, for sufficiently large values of $\|x\|$, we can assume that

$$\max\left\{0, \frac{2}{\varepsilon}(\|x\|^2 - a^2) + \lambda(x)\right\} = \frac{2}{\varepsilon}(\|x\|^2 - a^2) + \lambda(x).$$

By simple calculation, the expression of the penalty function becomes, in this case,

$$P(x) = \frac{1}{2}x^T Qx + c^T x + \frac{1}{\varepsilon}\left(\|x\|^2 - a^2\right)^2 + \lambda(x)\left(\|x\|^2 - a^2\right),$$

and the following inequalities hold:

$$P(x) \geq \left(-\frac{1}{2a^2}\|Q\| + \frac{1}{\varepsilon}\right)\|x\|^4 - \frac{\|c\|}{2a^2}\|x\|^3 - \left(\frac{2a^2}{\varepsilon} + \|Q\|\right)\|x\|^2 - \frac{3}{2}\|c\|\|x\| + \frac{a^4}{\varepsilon}.$$

As ε satisfies (13), we have that $\varepsilon \leq 2a^2/\|Q\|$ and hence we get $\lim_{\|x\| \rightarrow \infty} P(x) = \infty$. The existence of the global minimizer immediately follows from the continuity of P and the compactness of its level sets. $\square$

Now we state the first result about the exactness properties of the penalty function P . Since its proof is technical and lengthy we report it in the Appendix.

PROPOSITION 4.3. *A point $\bar{x} \in \mathbb{R}^n$ is a stationary point of $P(x)$ if and only if $(\bar{x}, \lambda(\bar{x}))$ is a KKT pair for problem (2). Furthermore, at this point we have $P(\bar{x}) = q(\bar{x})$.*

Now we prove that there is a one to one correspondence between global minimizers of problem (2) and global minimizers of the penalty function P .

PROPOSITION 4.4. *Every global minimizer of problem (2) is a global minimizer of $P(x)$, and vice versa.*

Proof. By Proposition 4.3, the penalty function P admits a global minimizer $\hat{x}$, which is obviously a stationary point of P , and hence by the preceding proposition we have that

$$P(\hat{x}) = q(\hat{x}).$$

On the other hand, if x^* is a global minimizer of problem (2), it is also a KKT point, and hence the preceding proposition implies again that $P(x^*) = q(x^*)$. Now we proceed by contradiction. Assume that a global minimizer $\hat{x}$ of $P(x)$ is not a global minimizer of problem (2), then there should exist a point x^* , global minimizer of problem (2), such that

$$P(\hat{x}) = q(\hat{x}) > q(x^*) = P(x^*)$$

that contradicts the assumption that $\hat{x}$ is a global minimizer of P . The converse is true by analogous considerations. $\square$

In order to complete the correspondence between the solution of problem (2) and the unconstrained minimization of the penalty function P we prove the following result that considers the correspondence between local minimizers.

PROPOSITION 4.5. *The function $P(x)$ admits at most one local-nonglobal minimizer $\bar{x}$. This local-nonglobal minimizer is the local-nonglobal minimizer of problem (2), and $\lambda(\bar{x})$ is the associated KKT multiplier.*

Proof. We first recall that if $\bar{x}$ is a local minimizer of $P(x)$ then the pair $(\bar{x}, \lambda(\bar{x}))$ satisfies the KKT conditions for problem (2). Moreover, by Proposition 4.3, we have that $P(\bar{x}) = q(\bar{x})$ and hence, since $\bar{x}$ is a local minimizer of P , there exists a neighborhood $\Omega(\bar{x})$ of $\bar{x}$ such that

$$q(\bar{x}) = P(\bar{x}) \leq P(x) \quad \text{for all } x \in \Omega(\bar{x}).$$

Thus, by using (iv) of Proposition 4.2, we obtain

$$(14) \quad q(\bar{x}) \leq P(x) \leq q(x) \quad \text{for all } x \in \Omega(\bar{x}) \cap \mathcal{F},$$

and hence $\bar{x}$ is a local minimizer for problem (2). The proof can be easily completed by recalling Proposition 2.2. $\square$

5. A new second order optimality condition. The results given in section 3 and section 4 can be combined to state new theoretical properties of problem (2). In this section we introduce a new second order necessary optimality condition for problem (2) for the nonconvex case that follows from the unconstrained formulation.

PROPOSITION 5.1. *Assume that Q is not positive semidefinite. If $\bar{x}$ is a global (local) minimizer of problem (2), then there exists a unique $\bar{\lambda} > 0$ such that the KKT conditions (4) hold and*

$$Q + 2\bar{\lambda}I + \frac{1}{a^2}(c\bar{x}^T + \bar{x}c^T) + \left(\frac{8}{a^2}\bar{\lambda} + \frac{2}{\varepsilon}\right)\bar{x}\bar{x}^T$$

is positive semidefinite for every ε satisfying (13).

Proof. If $\bar{x}$ is a global minimizer of problem (2), by Proposition 3.6 we have that $\bar{\lambda} > 0$ and, hence, $\|\bar{x}\|^2 = a^2$. Then there exists a neighborhood $\Omega(\bar{x})$ of $\bar{x}$ such that

$$\frac{2}{\varepsilon}(\|\bar{x}\|^2 - a^2) + \lambda(\bar{x}) \neq 0.$$

Thus, by (ii) of Proposition 4.2, the function $P(x)$ is twice continuously differentiable in $\Omega(\bar{x})$, and the Hessian matrix evaluated at $\bar{x}$ is given by

$$\nabla^2 P(\bar{x}) = Q + 2\lambda(\bar{x})I + \frac{1}{a^2}(c\bar{x}^T + \bar{x}c^T) + \left(\frac{8}{a^2}\lambda(\bar{x}) + \frac{2}{\varepsilon}\right)\bar{x}\bar{x}^T.$$

By Proposition 4.4, $\bar{x}$ is also a global minimizer of $P(x)$ and therefore $\bar{x}$ satisfies the second order necessary conditions to be a global unconstrained minimizer of P , that is $\nabla^2 P(\bar{x})$ is positive semidefinite. Then the result follows. $\square$

Recalling point (a) of Proposition 3.4, we have that in a global minimizer $\bar{x}$, it results in $c^T \bar{x} \leq 0$. Hence the matrix

$$\frac{1}{a^2}(c\bar{x}^T + \bar{x}c^T) + \left(\frac{8}{a^2}\bar{\lambda} + \frac{2}{\varepsilon}\right)\bar{x}\bar{x}^T$$

is not necessarily positive semidefinite. A similar second order necessary condition was given in [1], where it has been proved, without requiring any assumptions on the matrix Q , that if the global minimum is on the boundary, the matrix

$$\nabla^2 L(\bar{x}, \bar{\lambda}) + \frac{1}{a^2}(c\bar{x}^T + \bar{x}c^T) - \frac{1}{a^4}(c^T \bar{x})\bar{x}\bar{x}^T$$

is positive semidefinite where again the matrix $(c\bar{x}^T + \bar{x}c^T)/a^2 - (c^T \bar{x})\bar{x}\bar{x}^T/a^4$ is not necessarily positive semidefinite.

6. Algorithmic application. Besides their own theoretical interest, the results of the preceding sections are appealing also from a computational point of view. Although the study of a numerical algorithm for the solution of problem (2) is out of the aim of this paper, in this section we give a hint of possible algorithmic applications of the results of section 3 and section 4. We recall that Proposition 3.4 ensures that given a KKT point which is not a global solution for problem (2), it is possible to find a new feasible point with a lower value of the objective function and that Proposition 3.2 states that the number of KKT points with different values of the objective function is finite. These results indicate a new possibility to tackle large scale trust region problems. In fact they show that a global minimizer of problem (2) could be efficiently computed by applying a finite number of times a constrained optimization algorithm that presents the following features:

- (i) given a feasible starting point, it is able to locate a KKT point with a lower value of the objective function;
- (ii) it presents a “good” (at least superlinearly) rate of convergence;
- (iii) it does not require a heavy computational burden.

A possibility to ensure property (i) is to use any feasible method that forces the decrease of the objective function, following, for example, the approach of [37, 17]. Another possibility is to exploit the unconstrained reformulation of problem (2) described in section 4 which allows us to use any unconstrained method for the minimization of

the penalty function P . In fact, starting from a point x_0 , any unconstrained algorithm obtains a stationary point $\bar{x}$ for P such that

$$P(\bar{x}) < P(x_0).$$

Then Proposition 4.3 ensures that $\bar{x}$ is a KKT point of problem (2) and that $P(\bar{x}) = q(\bar{x})$. On the other hand, if x_0 is a feasible point, part (iv) of Proposition 4.2 yields that

$$q(\bar{x}) = P(\bar{x}) < P(x_0) \leq q(x_0).$$

In conclusion, by using an unconstrained algorithm we get a KKT point of problem (2) with a value of the objective function lower than the value at the starting point. Furthermore, the possibility of transforming the trust region problem into an unconstrained one seems to be quite appealing also as regards properties (ii) and (iii). In fact, Proposition 3.6 and (iii) of Proposition 4.2 guarantee that, in the nonconvex case, the penalty function is twice continuously differentiable in every local and global minimizer of the problem. Therefore, in this case, any unconstrained truncated Newton algorithm (see, for example, [5, 37, 15]) can be easily adapted in order to define globally convergent methods which show a superlinear rate of convergence in a neighborhood of every global or local minimizer. Nevertheless, we can define algorithms with superlinear rate of convergence without requiring that the penalty function is twice continuously differentiable in the neighborhood of the points of interest; that is without requiring the strict complementarity in these points. In fact we can draw our inspiration from the results in [8]. In particular, we can define a search direction d_k as follows:

$$(15) \quad \begin{aligned} & \text{(i) If } \|x_k\|^2 - a^2 \geq -\frac{\varepsilon}{2}\lambda(x_k), \\ & \begin{pmatrix} Q + 2\lambda(x_k)I & 2x_k \\ 2x_k^T & 0 \end{pmatrix} \begin{pmatrix} d_k \\ z_k \end{pmatrix} = - \begin{pmatrix} Qx_k + c \\ \|x_k\|^2 - a^2 \end{pmatrix}. \end{aligned}$$

$$(16) \quad \begin{aligned} & \text{(ii) If } \|x_k\|^2 - a^2 < -\frac{\varepsilon}{2}\lambda(x_k), \\ & (Q + 2\lambda(x_k)I) d_k = -(Qx_k + c). \end{aligned}$$

The results of [8] ensure that the algorithm $x_{k+1} = x_k + d_k$ is locally superlinearly convergent without requiring the strict complementarity. Following the approach of the truncated Newton method (see, for example, [5, 15]), in [24] it is shown that an approximate solution $\tilde{d}_k$ of (15), (16) is able to preserve the local superlinear rate of convergence of the algorithmic scheme. Furthermore, it is also proved that this direction $\tilde{d}_k$ satisfies suitable descent conditions with respect to the penalty function P . This strict connection between the direction $\tilde{d}_k$ and the penalty function $P(x)$ allows us to define globally and superlinearly convergent algorithms of the type

$$(17) \quad x_{k+1} = x_k + \alpha_k \tilde{d}_k,$$

where α_k can be determined by every stabilization technique and $\tilde{d}_k$ is computed by using a conjugate gradient based iterative method for solving approximately the linear system (15), (16).

The paper [24] is devoted to a complete description of this approach with the analysis of its theoretical properties and to the definition of an efficient algorithm.

TABLE 1
 Average number of iterations.

	First Set		Second Set	
n	Easy Case	Hard Case	Easy Case	Hard Case
100	11.3	21.9	10.7	25.6
256	11	23	11.9	27.9
324	12.9	23.6	12.9	29.9

 TABLE 2
 Average number of iterations.

Near Hard Case			
mult. $\lambda_{\min}$	1	5	10
$n = 100$	8.8	10.7	9.9
$n = 256$	9.1	6.9	9.1
$n = 324$	11	9.4	9.2

Here, in order to have only a preliminary idea of the viability of this unconstrained approach for solving problem (2), we have performed some numerical experiments with a rough implementation of algorithm (17) where α_k is determined by the line-search technique of [14] and $\tilde{d}_k$ is computed by a conjugate gradient algorithm similar to the one proposed in [5]. We coded the algorithm in MATLAB and ran the experiments on an IBM/RISC 6000. We ran two sets of randomly generated problems that we took from the collection of [34]. We solved ten related problems for each of the two classes both with the easy and the hard case. According to [34], the hard case occurs when the vector c is orthogonal to the subspace generated by the smallest eigenvalue of the matrix Q . In Table 1 we report the results in terms of average number of iterations for problems with increasing dimension ($n = 100, 256, 324$). We also ran a set of near hard-case problems (with $n = 100, 256, 324$), that is with c nearly orthogonal to the subspace of the smallest eigenvalue of Q . The results are reported in Table 2. We tested the invariance with respect to the multiplicity of the smallest eigenvalue (mult. of $\lambda_{\min} = 1, 5, 10$). The number of iterations is almost constant when the dimension increases. This feature is appealing when solving large scale problems taking into account that, at each iteration, the main effort is due to the approximate solution of a linear system of dimension n or $n - 1$ that requires only matrix-vector products. Furthermore, the efficiency of the algorithm seems not to be seriously affected by the occurrence of the hard case, while it is completely insensitive to the near-hard case. Of course, even if no final conclusion can be drawn by these limited numerical experiments, the results obtained encourage further research in defining new algorithms for solving large scale trust region problems which use the results described in this paper. In particular, as we said before, the possibility of defining efficient algorithms based on the unconstrained reformulation is investigated in [24].

Appendix. Before giving the proof of Proposition 4.3, we report this well-known result whose proof is immediate.

LEMMA A.1.

$$\max \left\{ \|\bar{x}\|^2 - a^2, -\frac{\varepsilon}{2} \lambda(\bar{x}) \right\} = 0$$

if and only if

$$\|\bar{x}\|^2 \leq a^2, \quad \lambda(\bar{x}) \geq 0, \quad \lambda(\bar{x})(\|\bar{x}\|^2 - a^2) = 0.$$

Proof of Proposition 4.3. First, we note that we can rewrite the gradient of P as follows

$$(18) \quad \begin{aligned} \nabla P(x) = & \nabla q(x) + 2\lambda(x)x + \nabla\lambda(x) \max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \right\} \\ & + \frac{4}{\varepsilon}x \max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \right\}. \end{aligned}$$

If part. Assume that $(\bar{x}, \bar{\lambda})$ is a KKT pair for problem (2), then, by recalling Lemma A.1, the result easily follows from the expression (18) of the gradient $\nabla P(x)$.

Only if part. In order to prove this part, it is enough to show that

$$(19) \quad \max \left\{ \|\bar{x}\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(\bar{x}) \right\} = 0.$$

In fact, from the expression (18) of the gradient of P , we have immediately that

$$0 = \nabla P(\bar{x}) = \nabla q(\bar{x}) + 2\lambda(\bar{x})\bar{x},$$

which together with Lemma A.1 implies that the point $(\bar{x}, \lambda(\bar{x}))$ is a KKT point of problem (2). We turn to prove (19). We distinguish the cases $\bar{x} = 0$ and $\bar{x} \neq 0$. The point $\bar{x} = 0$ is a stationary point for P if and only if $c = 0$. On the other hand, the point $\bar{x} = 0$ is a KKT point for problem (2) if and only if $c = 0$.

Now we assume $\bar{x} \neq 0$. By (18) we can write for every $x \neq 0$

$$\begin{aligned} \varepsilon x^T \nabla P(x) &= \varepsilon x^T \nabla q(x) + 2\lambda(x) \|x\|^2 + 4 \|x\|^2 \max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \right\} \\ &\quad + \varepsilon x^T \nabla \lambda(x) \max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \right\} \\ &= 2\varepsilon\lambda(x) \left(\|x\|^2 - a^2 \right) + \left(\varepsilon x^T \nabla \lambda(x) + 4 \|x\|^2 \right) \max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \right\}. \end{aligned}$$

Hence, taking into account that

$$\begin{aligned} 2\varepsilon\lambda(x) \left(\|x\|^2 - a^2 \right) &= (2\varepsilon\lambda(x) - 4(\|x\|^2 - a^2)) \max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \right\} \\ &\quad + 4 \max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \right\}^2, \end{aligned}$$

it easily follows that

$$(20) \quad \varepsilon x^T \nabla P(x) = \max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \right\} M(x; \varepsilon),$$

where

$$(21) \quad M(x; \varepsilon) = \varepsilon x^T \nabla \lambda(x) + 2\varepsilon\lambda(x) + 4a^2 + 4 \max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \right\}.$$

Now our aim is to show that $M(x; \varepsilon)$ is strictly positive for every $x \in \mathbb{R}^n$.

First, we consider the case $\max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \right\} = \|x\|^2 - a^2$; that is,

$$\|x\|^2 - a^2 \geq \frac{\varepsilon}{4a^2} (x^T Qx + c^T x).$$

By simple calculation we get the inequality

$$(22) \quad \|x\|^2 \geq \frac{8a^4 - \varepsilon\|c\|^2}{8a^2 + \varepsilon(2\|Q\| + 1)}.$$

In this case we have

$$\begin{aligned} M(x; \varepsilon) &= 4 \|x\|^2 + \varepsilon x^T \nabla \lambda(x) + 2\varepsilon \lambda(x) = 4 \|x\|^2 - \frac{\varepsilon}{2a^2} (4x^T Qx + 3c^T x) \\ &\geq 4 \|x\|^2 - \frac{\varepsilon}{2a^2} \left[4 \|Q\| \|x\|^2 + \frac{3}{2} (\|x\|^2 + \|c\|^2) \right] \\ &= \frac{1}{4a^2} \left[\|x\|^2 (16a^2 - \varepsilon(8\|Q\| + 3)) - 3\|c\|^2 \varepsilon \right]. \end{aligned}$$

Since ε satisfies (13), the term $16a^2 - \varepsilon(8\|Q\| + 3)$ is positive, and, by using (22), we can write the following inequality:

$$(23) \quad M(\bar{x}; \varepsilon) \geq \frac{\varepsilon^2 \|c\|^2 \|Q\| - 4a^2 p_1 \varepsilon + 64a^6}{2a^2 (8a^2 + \varepsilon(2\|Q\| + 1))},$$

where

$$(24) \quad p_1 = a^2(8\|Q\| + 3) + 5\|c\|^2.$$

The numerator of the right term of (23) is a quadratic function in ε which assumes positive values in the interval $(0, \varepsilon_1)$, where

$$\varepsilon_1 = b \left[p_1 - (p_1^2 - q)^{\frac{1}{2}} \right]$$

with

$$(25) \quad q = 16a^2 \|c\|^2 \|Q\|, \quad b = \frac{2a^2}{\|c\|^2 \|Q\|}.$$

Now we note the following relationships:

$$\varepsilon_1 = \frac{bq}{p_1 + (p_1^2 - q)^{\frac{1}{2}}} \geq \frac{bq}{2p_1} = \frac{16a^4}{a^2(8\|Q\| + 3) + 5\|c\|^2},$$

which, by the choice (13) for the parameter ε , imply that $\varepsilon \in (0, \varepsilon_1)$. Therefore, for all x such that $\max \{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \} = \|x\|^2 - a^2$ and for all ε satisfying (13), we get

$$M(x; \varepsilon) > 0.$$

Now let us consider the case $\max \{ \|x\|^2 - a^2, -\frac{\varepsilon}{2}\lambda(x) \} = -\frac{\varepsilon}{2}\lambda(x)$; that is,

$$\|x\|^2 - a^2 \leq \frac{\varepsilon}{4a^2} (x^T Qx + c^T x).$$

By simple calculation, and by the fact that ε satisfies (13), we have that $8a^2 - \varepsilon(2\|Q\| + 1)$ is positive and hence we have the inequality

$$(26) \quad \|x\|^2 \leq \frac{\varepsilon \|c\|^2 + 8a^4}{8a^2 - \varepsilon(2\|Q\| + 1)}.$$

In this case we have that

$$\begin{aligned} (27) \quad M(x; \varepsilon) &= \varepsilon x^T \nabla \lambda(x) + 4a^2 = -\frac{\varepsilon}{2a^2} (2x^T Qx + c^T x) + 4a^2 \\ &\geq -\frac{\varepsilon}{4a^2} \left[(4\|Q\| + 1) \|x\|^2 + \|c\|^2 \right] + 4a^2 \\ &\geq \frac{-\|c\|^2 \|Q\| \varepsilon^2 - 4a^2 p_2 \varepsilon + 64a^6}{2a^2 (8a^2 + \varepsilon(2\|Q\| + 1))}, \end{aligned}$$

where

$$p_2 = a^2 (8 \|Q\| + 3) + \|c\|^2.$$

The numerator of the last term of (27) is a quadratic function in ε which assumes positive values in the interval $(0, \varepsilon_2)$, where

$$\varepsilon_2 = b \left[-p_2 + (p_2^2 + q)^{\frac{1}{2}} \right]$$

and q, b are already defined by (25). Let us observe that $p_1, p_2 > 0$, $p_1 = p_2 + 4 \|c\|^2$, where p_1 is given by (24), and it is easily seen that $p_2^2 + q \leq p_1^2$; hence the following inequality holds:

$$-p_2 + (p_2^2 + q)^{\frac{1}{2}} = \frac{q}{p_2 + (p_2^2 + q)^{\frac{1}{2}}} \geq \frac{q}{2p_1}.$$

Now the choice (13) for the parameter ε implies that $\varepsilon \in (0, \varepsilon_2)$, and hence for every x such that $\max \{ \|x\|^2 - a^2, -\frac{\varepsilon}{2} \lambda(x) \} = -\frac{\varepsilon}{2} \lambda(x)$ we have

$$M(x; \varepsilon) > 0.$$

Therefore, if $\bar{x}$ is a point such that $\nabla P(\bar{x}) = 0$, recalling (20) we necessarily have that $\max \{ \|\bar{x}\|^2 - a^2, -\frac{\varepsilon}{2} \lambda(\bar{x}) \} = 0$, that is, equivalent to the conditions (19). Hence we have proved the first part of the proposition.

Now in order to complete the proof we have to show that $P(\bar{x}) = q(\bar{x})$. This easily follows from Lemma A.1, Proposition 4.1 (ii), and by observing that the penalty function can be rewritten as

$$P(x) = q(x) + \lambda(x) \max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2} \lambda(x) \right\} + \frac{1}{\varepsilon} \max \left\{ \|x\|^2 - a^2, -\frac{\varepsilon}{2} \lambda(x) \right\}^2. \quad \square$$

Acknowledgments. We wish to thank S. Santos, D. Sorensen, F. Rendl, and H. Wolkowiz for providing us with their MATLAB codes and test problems. We also thank the anonymous referees for their helpful suggestions which led to improvement of the paper.

REFERENCES

- [1] A. BAGCHI AND B. KALANTARI, *New Optimality Conditions and Algorithms for Homogeneous and Polynomial Optimization over Spheres*, Tech. report 40-90, RUTCOR, Rutgers University, New Brunswick, NJ, 1990.
- [2] J. BUNCH AND B. PARLETT, *Direct methods for solving symmetric indefinite systems of linear equations*, SIAM J. Numer. Anal., 8 (1971), pp. 639–655.
- [3] T. F. COLEMAN AND C. HEMPEL, *Computing a trust region step for a penalty function*, SIAM J. Sci. Statist. Comput., 11 (1990), pp. 180–201.
- [4] J. K. CULLUM AND R. A. WILLOUGHBY, *Lanczos Algorithms for Large Symmetric Eigenvalue Computation*, Birkhäuser Boston, Cambridge, MA, 1985.
- [5] R. DEMBO AND T. STEihaug, *Truncated-Newton methods algorithms for large-scale unconstrained optimization*, Math. Programming, 26 (1983), pp. 190–212.
- [6] G. DI PILLO AND L. GRIPPO, *An exact penalty method with global convergence properties for nonlinear programming problems*, Math. Programming, 36 (1986), pp. 1–18.
- [7] G. DI PILLO AND L. GRIPPO, *Exact penalty functions in constrained optimization*, SIAM J. Control Optim., 27 (1989), pp. 1333–1360.
- [8] F. FACCHINEI AND S. LUCIDI, *Quadratically and superlinear convergent algorithms for the solution of inequality constrained optimization problems*, J. Optim. Theory Appl., 85 (1985), pp. 265–289.

- [9] R. E. FLETCHER, *A class of methods for nonlinear programming with termination and convergence properties*, in Integer and Nonlinear Programming, J. Abadie, ed., North-Holland, Amsterdam, 1979, pp. 157–173.
- [10] A. FORSGREN, P. GILL, AND W. MURRAY, *Computing modified Newton directions using a partial Cholesky factorization*, SIAM J. Sci. Comput., 16 (1995), pp. 139–150.
- [11] G. FORSYTHE AND G. GOLUB, *On the stationary values of a second-degree polynomial on the unit sphere*, J. Soc. Indust. Appl. Math., 13 (1965), pp. 1050–1068.
- [12] D. M. GAY, *Computing optimal locally constrained steps*, SIAM J. Sci. Statist. Comput., 2 (1981), pp. 186–197.
- [13] T. GLAD AND E. POLAK, *A multiplier method with automatic limitation of penalty growth*, Math. Programming, 17 (1979), pp. 140–155.
- [14] L. GRIPPO, F. LAMPARIELLO, AND S. LUCIDI, *A nonmonotone line search technique for Newton's method*, SIAM J. Numer. Anal., 23 (1986), pp. 707–716.
- [15] L. GRIPPO, F. LAMPARIELLO, AND S. LUCIDI, *A truncated Newton method with nonmonotone linesearch for unconstrained optimization*, J. Optim. Theory Appl., 60 (1989), pp. 401–419.
- [16] L. GRIPPO AND S. LUCIDI, *A differentiable exact penalty function for bound constrained quadratic programming problems*, Optimization, 22 (1991), pp. 557–578.
- [17] M. HEINKENSCHLOSS, *On the solution of a two ball trust region subproblem*, Math. Programming, 64 (1994), pp. 249–276.
- [18] M. HESTENES, *Multiplier and gradient methods*, J. Optim. Theory Appl., 4 (1969), pp. 303–320.
- [19] A. KAMATH AND N. KARMARKAR, *A continuous approach to compute upper bounds in quadratic maximization problems with integer constraints*, in Recent Advances in Global Optimization, C. A. Floudas and P. M. Pardalos, eds., Princeton University Press, Princeton, NJ, 1991, pp. 125–140.
- [20] S. KAPOOR AND P. VAIDYA, *Fast algorithms for convex quadratic programming and multicommodity flows*, in Proc. 18th Annual ACM Symp. Theory Comput., ACM, New York, 1986, pp. 147–159.
- [21] N. KARMARKAR, *An interior-point approach to NP-complete problems*, in Proc. Mathematical Programming Society Conference on Integer Programming and Combinatorial Optimization, University of Waterloo, Waterloo, ON, Canada, 1990, pp. 351–366.
- [22] N. KARMARKAR, M. G. C. RESENDE, AND K. RAMAKRISHNAN, *An interior point algorithm to solve computationally difficult set covering problems*, Math. Programming, 52 (1991), pp. 597–618.
- [23] W. LI, *Differentiable piecewise quadratic exact penalty functions for quadratic programs with simple bound constraints*, SIAM J. Optim., 6 (1996), pp. 299–315.
- [24] S. LUCIDI AND L. PALAGI, *A class of algorithms for large scale trust region problems*, Tech. report 36.96, Dipartimento di Informatica e Sistemistica, Università di Roma “La Sapienza,” Roma, Italy, 1996.
- [25] S. LUCIDI, L. PALAGI, AND M. ROMA, *Quadratic Programs with Quadratic Constraint: Characterization of KKT Points and Equivalence with an Unconstrained Problem*, Tech. report 24.94, Dipartimento di Informatica e Sistemistica, Università di Roma “La Sapienza,” Roma, Italy, 1994.
- [26] J. MARTÍNEZ, *Local minimizers of quadratic functions on Euclidean balls and spheres*, SIAM J. Optim., 4 (1994), pp. 159–176.
- [27] J. MARTÍNEZ AND S. A. SANTOS, *Trust region algorithms on arbitrary domains*, Math. Programming, 68 (1995), pp. 267–302.
- [28] J. MORÉ, *Generalization of the trust region problem*, Optimization Methods and Software, 2 (1993), pp. 189–209.
- [29] J. MORÉ AND D. SORENSSEN, *On the use of directions of negative curvature in a modified Newton method*, Math. Programming, 16 (1979), pp. 1–20.
- [30] J. MORÉ AND D. C. SORENSSEN, *Computing a trust region step*, SIAM J. Sci. Statist. Comput., 4 (1983), pp. 553–572.
- [31] P. M. PARDALOS, Y. YE, AND C.-G. HAN, *Algorithms for the solution of quadratic knapsack problems*, Linear Algebra Appl., 25 (1991), pp. 69–91.
- [32] M. J. D. POWELL, *A method for nonlinear constraints in minimization problem*, in Optimization, R. Fletcher, ed., Academic Press, New York, 1969, pp. 283–298.
- [33] F. RENDL AND H. WOLKOWICZ, *A Semidefinite Framework to Trust Region Subproblems with Applications to Large Scale Minimization*, Tech. report CORR Rep. 94-32, University of Waterloo, Department of Combinatorics and Optimization, Waterloo, ON, Canada, 1994.
- [34] S. A. SANTOS AND D. C. SORENSSEN, *A New Matrix-Free for the Large Scale Trust Region Subproblem*, Tech. report TR95-20, Rice University, Houston, TX, 1995.
- [35] D. C. SORENSSEN, *Newton's method with a model trust region modification*, SIAM J. Numer.

- Anal., 19 (1982), pp. 409–426.
- [36] D. C. SORENSEN, *Minimization of a Large Scale Quadratic Function Subject to an Ellipsoidal Constraint*, Tech. report TR94-27, Rice University, Houston, TX, 1994.
 - [37] P. TOINT, *Towards an efficient sparsity exploiting Newton method for minimization*, in *Sparse Matrices and Their Uses*, I. S. Duff, ed., Academic Press, London, 1981, pp. 57–88.
 - [38] S. A. VAVASIS, *Nonlinear Optimization*, Oxford University Press, London, 1991.
 - [39] S. A. VAVASIS AND R. ZIPPEL, *Proving Polynomial-Time for Sphere-Constrained Quadratic Programming*, Tech. report 90-1182, Department of Computer Science, Cornell University, Ithaca, NY, 1990.
 - [40] Y. YE, *A new complexity result on minimization of a quadratic function with a sphere constraint*, in *Recent Advances in Global Optimization*, C. A. Floudas and P. M. Pardalos, eds., Princeton University Press, Princeton, NJ, 1991, pp. 19–31.
 - [41] Y. YE, *On affine scaling algorithms for nonconvex quadratic programming*, *Math. Programming*, 56 (1992), pp. 285–300.
 - [42] Y. YE AND E. TSE, *An extension of Karmarkar's projective algorithm for convex quadratic programming*, *Math. Programming*, 44 (1989), pp. 157–179.

GLOBAL LINEAR AND LOCAL QUADRATIC CONVERGENCE OF A LONG-STEP ADAPTIVE-MODE INTERIOR POINT METHOD FOR SOME MONOTONE VARIATIONAL INEQUALITY PROBLEMS*

JIE SUN[†] AND GONGYUN ZHAO[‡]

Abstract. An interior point (IP) method is proposed to solve variational inequality problems for monotone functions and polyhedral sets. The method has the following advantages: 1. Given an initial interior feasible solution with duality gap μ_0 , the algorithm requires at most $O[n \log(\mu_0/\epsilon)]$ iterations to obtain an ϵ -optimal solution. 2. The rate of convergence of the duality gap is q -quadratic. 3. At each iteration, a long-step improvement is allowed. 4. The algorithm can automatically transfer from a linear mode to a quadratic mode to accelerate the local convergence.

Key words. polynomial complexity of algorithms, interior point methods, monotone variational inequality problems, rate of convergence

AMS subject classifications. 49J35, 65K10, 90C06, 90C15, 90C33

PII. S1052623496300015

1. Introduction. Given a function $F : R^n \rightarrow R^n$ and a nonempty polyhedral convex set $\mathcal{P} \subset R^n$, consider the problem of finding a vector $u^* \in \mathcal{P}$ such that

$$(1) \quad F(u^*)^T(u - u^*) \geq 0 \quad \forall u \in \mathcal{P},$$

where the superscript “ T ” stands for the transpose. Assume that F is continuously differentiable in a neighborhood of R_+^n (the nonnegative orthant of R^n) and is monotone, that is, the Jacobian $F'(u)$ of F satisfies

$$x^T F'(u)x \geq 0 \quad \forall u \in R_+^n, x \in R^n.$$

Problem (1) is then a monotone variational inequality problem (MVIP). Given the representation of $\mathcal{P}$ as $\{u | Au = b, u \geq 0\}$ for some $A \in R^{m \times n}$ and $b \in R^m$, the usual Karush–Kuhn–Tucker stationary conditions

$$(2) \quad Au^* = b, u^* \geq 0, v^* = F(u^*) - A^T s^*, v^* \geq 0, \text{ and } (u^*)^T v^* = 0$$

can be used as an equivalent reformulation of problem (1), where $(u^*, s^*, v^*) \in R^n \times R^m \times R^n$.

For convenience, we assume that the matrix A has full row rank and call the value $u^T v$ the *duality gap*. A triple (u, s, v) is called a *feasible solution* to (2) if u, s , and v satisfy all requirements of (2) except the last one. A solution is μ -*optimal* if it is feasible and $u^T v \leq \mu$. The monotonicity of F implies that for any pair of feasible solutions (u, s, v) and (u', s', v') , one has

$$(u - u')^T(v - v') \geq 0.$$

* Received by the editors March 6, 1996; accepted for publication (in revised form) November 20, 1996. This research was partially supported by National University of Singapore grant RP-930033.
<http://www.siam.org/journals/siopt/8-1/30001.html>

[†] Department of Decision Sciences, National University of Singapore, Singapore 119260 (fbasunj@nus.sg).

[‡] Department of Mathematics, National University of Singapore, Singapore 119260 (matzgy@nus.sg).

Problem (2) is quite general because it includes as special cases some convex programming problems, some convex-concave saddle point problems, the monotone complementarity problem, and some Nash equilibrium problems [10]. Solution techniques for the problem have been extensively studied in the literature, but only a few of them are based on interior point (IP) methodology; see [10, 12, 16, 19] for examples.

IP methods have been successful in solving linear programming and linear complementarity problems. It is therefore reasonable to expect IP methods to perform well in solving the MVIP because the latter problem is a natural extension of the former problems. Possible advantages of applying IP methodology to the MVIP include global polynomial-time convergence and local superlinear convergence; both are very desirable features in numerical optimization.

Of particular interest to us among possible IP methods for the MVIP is the long-step path-following algorithms developed by Kojima, Megiddo, and Mizuno [6], Kojima, Mizuno, and Yoshise [7], and Mizuno, Todd, and Ye [9], which were originally designed for linear programming problems. The long-step methods usually have an $O[n \log(\mu_0/\epsilon)]$ iteration complexity, namely that starting from an initial feasible solution of duality gap μ_0 , they find an ϵ -optimal solution in at most $O[n \log(\mu_0/\epsilon)]$ iterations. Moreover, unlike the short-step path-following algorithms, they allow the next iterate to be determined in a large neighborhood of the current iterate. Therefore they perform much better than short-step methods in practice. By our computational experiences, long-step methods are generally three to five times faster than short-step methods in solving quadratic saddle-point problems arising from stochastic optimization [13, 14].

In this paper we develop a quadratically convergent $O[n \log(\mu_0/\epsilon)]$ long-step interior point method for (2). In our analysis, the global convergence of the algorithm requires a certain condition on F called *the scaled Lipschitz condition* (SLC). As pointed out by Hertog et al. [2], this condition implies a self-concordance property of F with respect to the logarithmic barrier. Thus our analysis on global convergence is, generally speaking, within the framework provided by Nesterov and Nemirovskii [10] for monotone variational inequality problems. However, since we use the long-step strategy, the general theory does not simply specialize to the proposed algorithm. It is our belief that the proposed algorithm is the first IP method for problem (1) that uses the so-called largest neighborhood $\mathcal{N}_\infty^-(\beta)$ (see its definition (3) in the next section) with polynomial complexity. In our proof of the global convergence, a nonlinear change of the duality gap has to be estimated which makes the analysis differ from the linear cases.

An attractive feature of the proposed algorithm is its quadratic rate of convergence. We show that, under simple assumptions such as the existence of a strictly complementary solution and the invertibility of a certain basic matrix, the duality gap converges q -quadratically and the sequence $\{u^k, s^k, v^k\}$ converges r -quadratically.

In order to achieve superlinear rate of convergence, it is typical that an IP method has to change its search directions when the iterates are sufficiently close to the optimal solution. However, the switching time from the regular direction to the accelerating direction is shown to exist only in theory and is often not known in a practical implementation. To solve this problem in the context of nonlinear MVIP, Ralph and Wright [12, 18] and Tseng [17] suggest some novel strategies. In the algorithm of Ralph and Wright every iteration consists of a fast step and a safe step; if the fast step fails, the algorithm reverts to the safe step. The method of Tseng uses a convex combination of the fast and the safe directions in the spirit of Gonzaga's

longest-step method for the linear complementarity problem [3]. In this paper we offer a different approach. Our algorithm consists of two modes, a linear mode and a quadratic mode. At each iteration the algorithm automatically selects a mode without reverting from the current computation. We show that the algorithm will permanently switch to the quadratic mode when the duality gap is small. Thus, the proposed algorithm combines all: the long-step improvement, the self-adaptive mode change, the polynomial complexity, and the quadratic rate of convergence. To make the paper concise, we consider a “feasible version” of the algorithm, that is, we assume an initial interior feasible solution is available.

We shall adopt the standard notations used in the literature of IP methods. For example, for any vector x , the corresponding capital letter X will denote the diagonal matrix whose (i, i) th entry is given by the i th component of x (denoted by x_i). In addition, for any real number ν , x^ν (X^ν , respectively) will represent the vector (the diagonal matrix) whose i th $((i, i)$ th) entry is x_i^ν . If a vector, a scalar or a matrix has to carry an index, then we put the index as a superscript in the case of vectors and as a subscript for scalars and matrices. The Euclidean norm and the infinity norm of x will be $\|x\|$ and $\|x\|_\infty$, respectively. Also, we shall use the letter e to stand for the vector of ones: $e = (1, 1, \dots, 1)^T$.

2. The algorithm. Let R_{++}^n be the positive orthant of R^n , and let $\mathcal{N}_\infty^-(\beta)$ be a set in $R_{++}^n \times R_{++}^n$ defined by

$$(3) \quad \mathcal{N}_\infty^-(\beta) := \left\{ (u, v) \mid Uv \geq \beta \mu e, \text{ where } \mu = u^T v / n \right\}.$$

The constant $\beta \in (0, 1)$ controls the size of the neighborhood $\mathcal{N}_\infty^-(\beta)$. At the extreme cases, if $\beta = 0$ then the neighborhood is the entire positive orthant and, if $\beta = 1$, the neighborhood shrinks to the so-called central path. Notice that if $v = F(u) - A^T s$, $Au = b$, $(u, v) \in \mathcal{N}_\infty^-(\beta)$, and $\mu \leq \epsilon$, then (u, v, s) is an $O(\epsilon)$ -solution to (2) in the sense

$$v = F(u) - A^T s, \quad Au = b, \quad u \geq 0, \quad v \geq 0, \quad u^T v \leq n\epsilon.$$

Given an interior feasible solution (u, v, s) and a number ζ , we define the search direction by the following system:

$$(4) \quad \begin{cases} F'(u)\Delta u - A^T \Delta s - \Delta v = 0, \\ A\Delta u = 0, \\ V\Delta u + U\Delta v = -Vu + \zeta e. \end{cases}$$

Due to the monotonicity of F and the full-rank assumption on A , this system always has a unique solution $(\Delta u, \Delta s, \Delta v)$. Suppose that an initial feasible point $(u^0, v^0) \in \mathcal{N}_\infty^-(\beta)$ is available. The proposed algorithm proceeds as follows.

ALGORITHM.

- **Step 0 (Initialization).** Let $\epsilon > 0$ be a given tolerance, let M be a scaled Lipschitz constant defined in (6), and let $\beta, \lambda \in (0, 1)$ be constants. Set $k = 0$, $\theta_0 = 0$, and $\mu_0 = (u^0)^T v^0 / n$.
- **Step 1 (Computing the Search Direction).** If $\mu_k < 1$ and $\theta_k = 1$, then go to Step 1.2; otherwise, go to Step 1.1.

Step 1.1 (The Linear Mode). Solve the equation system (4) with $(u, v, s) = (u^k, v^k, s^k)$ and $\zeta = \lambda \mu_k$. Go to Step 2.

Step 1.2 (The Quadratic Mode). If at the last iteration the algorithm was in linear mode, set $\alpha = \lambda / \mu_k$; otherwise, keep the current α . Solve the equation system (4) with $(u, v, s) = (u^k, v^k, s^k)$ and $\zeta = \alpha \mu_k^2$. Go to Step 2.

• **Step 2 (The Line Search).** Let

$$\begin{cases} u(\theta) = u + \theta \Delta u, \\ s(\theta) = s + \theta \Delta s, \\ v(\theta) = F(u(\theta)) - A^T s(\theta), \\ \mu(\theta) = u(\theta)^T v(\theta)/n. \end{cases}$$

If $(u(1), v(1)) \in \mathcal{N}_\infty^-(\beta)$ and $\mu(1) \leq (1-1/n)\mu_k$, then set $\theta_{k+1} = 1$. Otherwise, determine θ_{k+1} by

$$(5) \quad \mu(\theta_{k+1}) \leq \mu(\hat{\theta}) \text{ for all } \theta \in [\hat{\theta}, 1) \text{ and } (u(\theta), v(\theta)) \in \mathcal{N}_\infty^-(\beta),$$

where

$$\hat{\theta} = \frac{(1-\beta)\lambda}{(1+\beta)(1+M)n}.$$

• **Step 3 (Optimality Test).** Set

$$(u^{k+1}, s^{k+1}, v^{k+1}, \mu_{k+1}) = (u(\theta_{k+1}), s(\theta_{k+1}), v(\theta_{k+1}), \mu(\theta_{k+1})).$$

If $\mu_{k+1} \leq \epsilon$, stop; otherwise, set $k := k+1$ and go to Step 1.

The constant ζ in the algorithm determines the search direction. At the extreme cases, if $\zeta = 0$, the direction is the Newton direction for the problem $Au = b$, $v = F(u) - A^T s$, $u^T v = 0$, which is called *the affine scaling direction* in the terminology of IP methods; if $\zeta = \mu_k$, the direction is the so-called *centering direction*. The fact $\lambda \in (0, 1)$ indicates that the search direction in the linear mode is between the affine scaling and the centering directions. In the quadratic mode, the search direction is “quadratically approaching” to the affine scaling direction so as to achieve quadratic convergence. This idea can be traced back to the genuine quadratic algorithm of Luo and Ye [8] for linear programming. What appears to be new here is the simple criterion for transferring to the quadratic mode (i.e., $\mu_k < 1$ and $\theta_k = 1$). The algorithm starts from the linear mode, perhaps occasionally jumps between the linear and the quadratic modes, and, as we shall show, eventually switches to the quadratic mode when μ_k is small enough.

The line search at Step 2 for the determination of θ_{k+1} is flexible for implementation. The straightforward choice is to set $\theta_{k+1} = 1$ or $\hat{\theta}$ at all iterations. However, for fast convergence, a line search on $[\hat{\theta}, 1]$ is recommended.

3. Analysis of the polynomial complexity. The purpose of this section is to obtain an $O[n \log(\mu_0/\epsilon)]$ estimate for the iteration complexity of the algorithm, where μ_0 is the initial duality gap defined in Step 0 of the algorithm. For a given iteration of the algorithm, if the search direction is computed by Step 1.1 (the linear mode), then we call this iteration a *linear iteration*; otherwise, we call it a *quadratic iteration*. The analysis is made for problems that satisfy the following SLC.

SLC Assumption. There exist constants $\gamma \in (0, 1)$ and $M > 0$ such that the inequality

$$(6) \quad \|U[F(u + \Delta u) - F(u) - F'(u)\Delta u]\|_\infty \leq 2M\Delta u^T F'(u)\Delta u$$

holds for any vectors $u > 0$ and Δu satisfying $\|U^{-1}\Delta u\| \leq \gamma$.

Similar conditions have been used in the literature for the analysis of IP methods for convex programming [21], saddle-point problems [15], the monotone complementarity problem [1, 11], the P^* -complementarity problem [5], and some variational inequality problems [16, 19]. Functions satisfying this condition include $F(u) = Qu + c$, where Q is a positive semidefinite matrix, and $F_j(u) = f_j(u_j)$, where $f_j(u_j)$, could be

$$\log u_j, \quad -u_j^\nu (\nu \leq -1), \quad \text{or} \quad u_j^\nu (\nu > 0).$$

Note that the set of functions satisfying SLC is a convex cone; e.g., any positive linear combination of such functions still satisfies SLC.

The following lemma shows that we can concentrate on the analysis of a linear iteration as far as the global convergence is concerned.

LEMMA 3.1. *If there exists a constant $0 < \Omega \leq 1$ such that at each linear iteration one has*

$$(7) \quad \mu_{k+1} \leq \left(1 - \frac{\Omega}{n}\right) \mu_k,$$

then it requires at most $k = O[n \log(\mu_0/\epsilon)]$ iterations to reduce μ_0 to ϵ .

Proof. If $\theta_{k+1} = 1$, then either in the linear or the quadratic mode of the algorithm we always have

$$(8) \quad \mu_{k+1} \leq \left(1 - \frac{1}{n}\right) \mu_k \leq \left(1 - \frac{\Omega}{n}\right) \mu_k.$$

If $\theta_{k+1} < 1$ in a quadratic iteration, then by the rule of the line search we have $\mu_{k+1} \leq \mu_k$ and the $k+1$ st iteration is a linear iteration. By the assumption of the lemma, we have

$$(9) \quad \mu_{k+2} \leq \left(1 - \frac{\Omega}{n}\right) \mu_{k+1} \leq \left(1 - \frac{\Omega}{n}\right) \mu_k.$$

The inequalities (7), (8), and (9) imply

$$\mu_{k+2} \leq \left(1 - \frac{\Omega}{n}\right) \mu_k \text{ for } k = 0, 1, \dots$$

Therefore we obtain

$$\mu_k \leq \left(1 - \frac{\Omega}{n}\right)^{\lfloor k/2 \rfloor} \mu_0,$$

where $\lfloor k/2 \rfloor$ represents the largest integer not more than $k/2$. The lemma follows from this. $\square$

Now we establish several lemmas (Lemmas 3.2–3.6) for the linear iterations of the algorithm. For simplicity of notations, the iteration index k is omitted in the following analysis. Let

$$h(\theta) = \theta^2 \Delta U \Delta v + (U + \theta \Delta U) [F(u + \theta \Delta u) - F(u) - \theta F'(u) \Delta u],$$

which represents the “nonlinear change” of μ from iteration k to iteration $k+1$ (see formula (10) below for the reason why it is so). The following is a key lemma for our complexity analysis. It shows two points: (i) If the size of $h(\theta)$ is of order $O(\theta\mu)$, then all iterates are in $\mathcal{N}_\infty^-(\beta)$. (ii) The constant Ω in (7) is of order $O(\theta)$.

LEMMA 3.2. Let $(u, v) \in \mathcal{N}_\infty^-(\beta)$. If

$$\|h(\theta)\|_\infty \leq \frac{1-\beta}{1+\beta} \theta \lambda \mu,$$

then

$$(u(\theta), v(\theta)) \in \mathcal{N}_\infty^-(\beta)$$

and

$$\mu(\theta) \leq \left(1 - \frac{1+\beta-2\lambda}{1+\beta} \theta\right) \mu.$$

Proof.

$$\begin{aligned} U(\theta)v(\theta) &= U(\theta)[F(u + \theta\Delta u) - A^T s - \theta A^T \Delta s] \\ &= (U + \theta\Delta U)[v + \theta\Delta v + F(u + \theta\Delta u) - F(u) - \theta F'(u)\Delta u] \\ &= Uv + \theta(U\Delta v + V\Delta u) + \theta^2 \Delta U \Delta v \\ &\quad + (U + \theta\Delta U)[F(u + \theta\Delta u) - F(u) - \theta F'(u)\Delta u] \\ &= Uv + \theta(-Uv + \lambda\mu e) + h(\theta) \\ (10) \quad &= (1 - \theta)Uv + \theta\mu\lambda e + h(\theta). \end{aligned}$$

Note that $(u, v) \in \mathcal{N}_\infty^-(\beta)$, so we have

$$\begin{aligned} U(\theta)v(\theta) &\geq (1 - \theta)\beta\mu e + \theta\lambda\mu e + h(\theta) \\ (11) \quad &\geq [(1 - \theta)\beta + \theta\lambda]\mu e - \|h(\theta)\|_\infty e \end{aligned}$$

and

$$\begin{aligned} \mu(\theta) &= e^T U(\theta)v(\theta)/n \\ &= (1 - \theta)u^T v/n + \theta\lambda\mu + e^T h(\theta)/n \\ &= (1 - \theta + \theta\lambda)\mu + e^T h(\theta)/n \\ (12) \quad &\leq (1 - \theta + \theta\lambda)\mu + \|h(\theta)\|_\infty. \end{aligned}$$

We get

$$\begin{aligned} &U(\theta)v(\theta) - \beta\mu(\theta)e \\ &\geq [(1 - \theta)\beta + \theta\lambda]\mu e - \|h(\theta)\|_\infty e - \beta(1 - \theta + \theta\lambda)\mu e - \beta\|h(\theta)\|_\infty e \\ &= (1 - \beta)\theta\lambda\mu e - (1 + \beta)\|h(\theta)\|_\infty e \geq 0. \end{aligned}$$

Therefore we have $(u(\theta), v(\theta)) \in \mathcal{N}_\infty^-(\beta)$.

From (12) and the assumption of this lemma it follows that

$$\mu(\theta) \leq \left[1 - \theta + \left(1 + \frac{1-\beta}{1+\beta}\right)\lambda\theta\right] \mu = \left(1 - \frac{1+\beta-2\lambda}{1+\beta}\theta\right) \mu. \quad \square$$

The following three lemmas are devoted to the estimation of $h(\theta)$ so that an explicit lower bound on θ can be derived (see $\hat{\theta}$ in the proof of Lemma 3.6). Among them, Lemma 3.3 is a straightforward extension of a standard result for monotone

linear complementarity problems (e.g., see [20]). To make the paper self-contained, however, we give a complete proof of it.

LEMMA 3.3. *Let*

$$D := U^{1/2}V^{-1/2} \quad \text{and} \quad w := (UV)^{-1/2}(-Uv + \lambda\mu e).$$

Then

$$(13) \quad \max \{ \|D^{-1}\Delta u\|, \|D\Delta v\| \} \leq \|w\|, \quad \|\Delta U \Delta v\| \leq \|w\|^2,$$

and

$$(14) \quad 0 \leq \Delta u^T \Delta v \leq \|w\|^2/4,$$

where by our convention ΔU represents a diagonal matrix whose diagonal elements are the components of Δu .

Proof. From (4) we have

$$D^{-1}\Delta u + D\Delta v = w.$$

Since $A\Delta u = 0$ and

$$(15) \quad \Delta u^T \Delta v = \Delta u^T F'(u)\Delta u - \Delta u^T A^T \Delta s = \Delta u^T F'(u)\Delta u \geq 0,$$

we have

$$\|D^{-1}\Delta u\|^2 + \|D\Delta v\|^2 \leq \|D^{-1}\Delta u + D\Delta v\|^2 = \|w\|^2,$$

which implies (13) and

$$\begin{aligned} \|\Delta U \Delta v\| &= \|\Delta U D^{-1} D\Delta v\| \leq \|D^{-1}\Delta u\| \|D\Delta v\| \\ &\leq \max \{ \|D^{-1}\Delta u\|^2, \|D\Delta v\|^2 \} \leq \|w\|^2. \end{aligned}$$

From (4) we obtain

$$D^{-2}\Delta u + \Delta v = -v + \lambda\mu u^{-1}$$

and

$$\Delta u^T D^{-2}\Delta u + \Delta u^T \Delta v = \Delta u^T (-v + \lambda\mu u^{-1}) = \Delta u^T D^{-1}w.$$

Hence

$$\begin{aligned} \Delta u^T \Delta v &= w^T D^{-1}\Delta u - \|D^{-1}\Delta u\|^2 \\ &\leq \|w\| \cdot \|D^{-1}\Delta u\| - \|D^{-1}\Delta u\|^2. \end{aligned}$$

Using the inequality

$$bt - t^2 \leq b^2/4,$$

where b and t are any real numbers, we obtain

$$\Delta u^T \Delta v \leq \|w\|^2/4. \quad \square$$

Using Lemma 3.3, we are now able to estimate $h(\theta)$ in terms of w .

LEMMA 3.4. *If $\theta \|U^{-1}\Delta u\| \leq \gamma$, then*

$$\|h(\theta)\|_\infty \leq (1 + M) \theta^2 \|w\|^2,$$

where γ and M are the constants defined in (6).

Proof. By the SLC assumption and noting that

$$|u_i + \theta \Delta u_i| \leq (1 + \gamma) u_i \leq 2u_i \quad \forall i = 1, \dots, n,$$

we have

$$\begin{aligned} \|h(\theta)\|_\infty &\leq \theta^2 \|\Delta U \Delta v\|_\infty + \|2U [F(u + \theta \Delta u) - F(u) - \theta F'(u) \Delta u]\|_\infty \\ &\leq \theta^2 \|\Delta U \Delta v\| + 4M\theta^2 \Delta u^T F'(u) \Delta u \\ &\leq \theta^2 \|w\|^2 + M\theta^2 \|w\|^2 \\ &= (1 + M) \theta^2 \|w\|^2. \end{aligned}$$

The last inequality above is due to (13), (15), and (14). $\square$

We now give a bound for w .

LEMMA 3.5. *Let $Uv \geq \beta \mu e$, and let $\lambda \leq 2\beta$. Then $\|w\|^2 \leq n\mu$.*

Proof. Let

$$r = \left(\frac{1}{\mu} UV\right)^{1/2} e \quad \text{and} \quad r^{-1} = \left(\frac{1}{\mu} UV\right)^{-1/2} e.$$

Then

$$\|w\|^2 = \mu \|-r + \lambda r^{-1}\|^2 = \mu [\|r\|^2 + \|\lambda r^{-1}\|^2 - 2\lambda n]$$

and

$$\|r\|^2 = \sum_{i=1}^n \frac{u_i v_i}{\mu} = \frac{1}{\mu} u^T v = n.$$

Since $u_i v_i \geq \mu \beta$, we have $r^{-1} \leq \beta^{-1/2} e$ and therefore $\|r^{-1}\|^2 \leq \beta^{-1} n$. Hence

$$\|w\|^2 \leq \mu (n + \lambda^2 \beta^{-1} n - 2\lambda n) \leq n\mu. \quad \square$$

Remark. The condition $\lambda \leq 2\beta$ of Lemma 3.5 is not essential, what we really need is $\|w\|^2 = O(n\mu)$.

LEMMA 3.6. *Suppose that F satisfies SLC. Let $0 < \lambda \leq 2\beta$ and $0 < \beta < 0.25\gamma$, where γ is defined in (6). If the k th iteration is linear, then*

$$\mu_{k+1} \leq \left(1 - \frac{\Omega}{n}\right) \mu_k,$$

where

$$\Omega := \frac{1 + \beta - 2\lambda}{1 + \beta} \frac{(1 - \beta)\lambda}{(1 + \beta)(1 + M)} > 0.$$

Proof. If $\theta_{k+1} = 1$, then according to the line search rule $\mu_{k+1} \leq (1 - 1/n)\mu_k$. If $\theta_{k+1} < 1$, consider

$$\hat{\theta} = \frac{(1 - \beta)\lambda}{(1 + \beta)(1 + M)n}$$

and note that

$$\begin{aligned} \|U^{-1}\Delta u\| &= \|(UV)^{-1/2}D^{-1}\Delta u\| \leq \frac{1}{\sqrt{\beta\mu}} \|D^{-1}\Delta u\| \\ &\leq \frac{1}{\sqrt{\beta\mu}} \|w\| \leq \sqrt{\frac{n}{\beta}}. \end{aligned}$$

The first inequality holds because $(u, v) \in \mathcal{N}_{\infty}^{-}(\beta)$. The last two inequalities are due to Lemmas 3.3 and 3.5, respectively. Thus $\hat{\theta} \|U^{-1}\Delta u\| \leq \gamma$. Now using Lemma 3.4, Lemma 3.5, and the definition of $\hat{\theta}$ we have

$$(16) \quad \|h(\hat{\theta})\|_{\infty} \leq (1 + M) \hat{\theta}^2 n \mu = \frac{1 - \beta}{1 + \beta} \lambda \hat{\theta} \mu.$$

By Lemma 3.2 we have

$$(u(\hat{\theta}), v(\hat{\theta})) \in \mathcal{N}_{\infty}^{-}(\beta).$$

By the rule of line searches of the algorithm we have $(u^{k+1}, v^{k+1}) \in \mathcal{N}_{\infty}^{-}(\beta)$ and

$$\mu_{k+1} \leq \mu(\hat{\theta}).$$

From Lemma 3.2 and (16) we have

$$\begin{aligned} \mu(\hat{\theta}) &\leq \left(1 - \frac{1 + \beta - 2\lambda}{1 + \beta} \hat{\theta}\right) \mu_k \\ &= \left[1 - \frac{1 + \beta - 2\lambda}{1 + \beta} \frac{(1 - \beta)\lambda}{(1 + \beta)(1 + M)n}\right] \mu_k. \end{aligned}$$

Since $0 < \lambda \leq 2\beta$ and $0 \leq \beta \leq 1/4$, there holds

$$\Omega = \frac{1 + \beta - 2\lambda}{1 + \beta} \frac{(1 - \beta)\lambda}{(1 + \beta)(1 + M)} > 0.$$

Since $\Omega \leq 1$, for both cases $\theta_{k+1} = 1$ and $\theta_{k+1} < 1$, we have

$$\mu_{k+1} \leq \left(1 - \frac{\Omega}{n}\right) \mu_k. \quad \square$$

Lemmas 3.6 and 3.1 imply the following main result on polynomial time complexity.

THEOREM 3.7. *Let $0 < \lambda \leq 2\beta$ and $0 < \beta < 0.25\gamma$, where γ is defined in (6). Then for any problem with F satisfying SLC, the algorithm needs no more than $O[n \log(\mu_0/\epsilon)]$ iterations to achieve $\mu_k \leq \epsilon$.*

4. Local quadratic rate of convergence. In this section we consider the convergence rate when μ is small. Our major results are Theorems 4.6 and 4.7. They state that there is a constant c_3 such that $\mu_{k+1} \leq c_3 \mu_k^2$ for all sufficiently large k and that the sequence $\{(u^k, s^k, v^k)\}$ converges to a solution of (2) r -quadratically.

A solution (u^*, s^*, v^*) to problem (2) is called *strictly complementary* if $u^* + v^* > 0$. Let

$$B := \{i \in \{1, 2, \dots, n\} | u_i^* > 0\} \quad \text{and} \quad N := \{1, 2, \dots, n\} \setminus B$$

and let u_B and u_N be the partition of u in terms of B and N . For the same B and N , let v_B and v_N be the corresponding partition of v . Since (u^*, s^*, v^*) is strictly complementary, we have $v_N^* > 0$. For ease of discussion, we split the Jacobian matrix $F'(u)$ into four blocks according to B and N :

$$F'(u) = \begin{bmatrix} M_{BB}(u) & M_{BN}(u) \\ M_{NB}(u) & M_{NN}(u) \end{bmatrix}.$$

Also, we partition $F(u) = (F_B(u)^T, F_N(u)^T)^T$ and $A = (A_B, A_N)$. It should be noted that the partition (B, N) above does not depend on any specific choice of the strictly complementary solution, as indicated by the following lemma.

LEMMA 4.1. *Let (u^*, s^*, v^*) be any strictly complementary solution and (B, N) be the corresponding partition of its index set. Then any solution (u, s, v) will necessarily have $u_N = 0$ and $v_B = 0$. Consequently, any strictly complementary solution will necessarily have $u_B > 0$ and $v_N > 0$.*

Proof. From monotonicity of F ,

$$0 \leq (u - u^*)^T (v - v^*) = -u^T v^* - v^T u^*.$$

Thus $u^T v^* = v^T u^* = 0$, which implies the conclusions. $\square$

Define

$$(17) \quad \eta := \min_{i \in B, j \in N} \{u_i^*, v_j^*\} \quad \text{and} \quad \xi := \frac{\eta \beta}{n}$$

and let

$$\mathcal{F} := \left\{ u \in R_+^n \mid u_B \geq \xi e, \ u_i \leq 2(u^0)^T v^0 / v_i^0 \quad \forall i = 1, \dots, n \right\},$$

where (u^0, v^0) is the initial solution in Step 0 of the algorithm in section 3.

We shall derive the quadratic rate of convergence for the algorithm under the following assumptions.

Assumption 1. The problem has a strictly complementary solution.

Assumption 2. The matrix

$$\begin{bmatrix} M_{BB}(u) & -A_B^T \\ A_B & 0 \end{bmatrix}$$

is invertible for every $u \in \mathcal{F}$.

Assumption 3. There exist a number $\delta > 0$ and a constant L such that for all $i \in B$

$$|F_i(u + \Delta u) - F_i(u) - F'_i(u) \Delta u| \leq L \|\Delta u\|^2 \quad \forall u \in \mathcal{F}, \|\Delta u\| \leq \delta.$$

Assumptions 1 and 3 are standard in the IP methodology. An example of Ye and Anstreicher shows that without Assumption 1 our algorithm may not have superlinear convergence [20]. Assumption 2 is new and can be implied by some stronger conditions such as A_B has full row rank plus the invertibility of the matrix $H^T M_{BB}(u)H$, where the columns of H form a basis for the null space of A_B .

The key factor in the analysis of the local convergence is to show that we have

$$\|\Delta u\| = O(\mu_k) \quad \text{and} \quad \|\Delta v\| = O(\mu_k),$$

where the big O constant is independent of k but possibly dependent on n and the problem data. The next technical lemma establishes certain bounds for some important quantities. For simplicity we drop the index k and use notations like U, V, u, s, v , and μ instead. Recall that

$$D := U^{1/2}V^{-1/2}.$$

LEMMA 4.2. *Let $\mu = u^T v / n$. Then*

$$(18) \quad 0 < u_i \leq 2(u^0)^T v^0 / v_i^0, \quad 0 < v_i \leq 2(u^0)^T v^0 / u_i^0, \quad i = 1, \dots, n,$$

and

$$(19) \quad \|D^{-1}\Delta u\| = O(\sqrt{\mu}), \quad \|D\Delta v\| = O(\sqrt{\mu}).$$

If Assumption 1 holds, then

$$(20) \quad u_i \geq \xi \quad \forall i \in B, \quad u_i \leq \frac{n}{\eta}\mu \quad \forall i \in N; \quad v_i \geq \xi \quad \forall i \in N, \quad v_i \leq \frac{n}{\eta}\mu \quad \forall i \in B;$$

where ξ is defined in (17).

Proof. From monotonicity of F , we have $(u - u^0)^T(v - v^0) \geq 0$, which, together with the fact that $\mu \leq \mu_0$, implies that

$$u^T v^0 + (u^0)^T v \leq (u^0)^T v^0 + u^T v \leq 2(u^0)^T v^0.$$

Thus we obtain (18).

We now prove (19). From (4),

$$D^{-1}\Delta u + D\Delta v = -(UV)^{-1/2}(Uv - \zeta e), \quad \text{and} \quad \Delta u^T \Delta v = \Delta u^T F'(u)\Delta u \geq 0,$$

we get

$$\begin{aligned} \|D^{-1}\Delta u\|^2 + \|D\Delta v\|^2 &\leq (\zeta e - Uv)^T (UV)^{-1} (\zeta e - Uv) \\ &\leq u^T v + \zeta^2 e^T (UV)^{-1} e \\ &\leq u^T v + n\zeta^2 / (\beta\mu) \\ &= O(\mu). \end{aligned}$$

To prove (20), we note that $(u, v) \in \mathcal{N}_\infty^-(\beta)$; i.e.,

$$\frac{\min_{1 \leq i \leq n} \{u_i v_i\}}{u^T v} \geq \frac{\beta}{n} > 0.$$

By Assumption 1, there is a strictly complementary solution (u^*, s^*, v^*) . Therefore recalling

$$\eta = \min_{i \in B, j \in N} \{u_i^*, v_j^*\} > 0$$

and using

$$(u^* - u)^T (v^* - v) \geq 0,$$

we have

$$\sum_{i \in B} u_i^* v_i + \sum_{i \in N} v_i^* u_i \leq u^T v.$$

Thus for all $i \in B$

$$u_i = \frac{u_i v_i}{v_i} \geq \frac{\beta(u^T v)}{n} \frac{u_i^*}{v_i u_i^*} \geq \frac{\beta u_i^*}{n} \geq \frac{\beta \eta}{n} = \xi.$$

For all $i \in N$

$$u_i \leq \frac{u^T v}{v_i^*} = \frac{n\mu}{v_i^*} \leq \frac{n\mu}{\eta}.$$

The proof for v_i is similar. Thus (20) is valid. $\square$

Inequalities (18) and (20) guarantee that for all k the iterate (u^k, v^k) remains in the compact set $\mathcal{F}$. In fact, these inequalities motivate the definition of $\mathcal{F}$. An implication of (20) is that any solution obtained by the algorithm must be a strictly complementary solution. We now have the following lemma.

LEMMA 4.3.

$$\|\Delta u_N\| = O(\mu) \quad \text{and} \quad \|\Delta v_B\| = O(\mu).$$

Proof. From (19) and (20), we get

$$\begin{aligned} \|\Delta u_N\| &= \|D_N D_N^{-1} \Delta u_N\| \\ &\leq \|D_N\| \|D_N^{-1} \Delta u_N\| \\ &= \|D_N\| O(\sqrt{\mu}) \\ &= \|(U_N V_N)^{1/2} V_N^{-1}\| O(\sqrt{\mu}) \\ &= \|(U_N V_N)^{1/2}\| O(1/\xi) O(\sqrt{\mu}) \\ &= O(\sqrt{\mu}) O(1/\xi) O(\sqrt{\mu}) = O(\mu). \end{aligned}$$

The proof for $\|\Delta v_B\| = O(\mu)$ is similar. $\square$

Remark. Lemmas 4.1–4.3 are direct extensions of the well-known results for monotone linear complementarity problems. Their original proofs are due to Güler and Ye [4] and Ye and Anstreicher [20].

LEMMA 4.4. *Under Assumptions 1 and 2, the solution $(\Delta u, \Delta v)$ of the system (2.1) satisfies*

$$\|\Delta u\| \leq c_1 \mu, \quad \|\Delta s\| \leq c_1 \mu, \quad \text{and} \quad \|\Delta v\| = c_1 \mu,$$

where c_1 is a certain constant independent of k .

Proof. From (4), we have

$$(21) \quad M_{BB}(u)\Delta u_B - A_B^T \Delta s = -M_{BN}(u)\Delta u_N + \Delta v_B,$$

$$(22) \quad M_{NB}(u)\Delta u_B - A_N^T \Delta s - \Delta v_N = -M_{NN}(u)\Delta u_N,$$

$$(23) \quad A_B u_B = -A_N \Delta u_N.$$

From (21), (23), and Assumption 2, we get

$$\begin{bmatrix} \Delta u_B \\ \Delta s \end{bmatrix} = \begin{bmatrix} M_{BB}(u) & -A_B^T \\ A_B & 0 \end{bmatrix}^{-1} \begin{bmatrix} -M_{BN}(u)\Delta u_N + \Delta v_B \\ -A_N \Delta u_N \end{bmatrix}.$$

Since $u \in \mathcal{F}$ and $\mathcal{F}$ is compact, the matrix

$$\begin{bmatrix} M_{BB}(u) & -A_B^T \\ A_B & 0 \end{bmatrix}^{-1}$$

is uniformly bounded on $\mathcal{F}$. The boundedness of $M_{BN}(u)$ and A_N then implies

$$(24) \quad \|\Delta u_B\| = O(\|\Delta u_N\|) + O(\|\Delta v_B\|) = O(\mu)$$

and

$$(25) \quad \|\Delta s\| = O(\|\Delta u_N\|) + O(\|\Delta v_B\|) = O(\mu).$$

Then by (22) and the boundedness of $M_{NB}(u)$ and $M_{NN}(u)$, we get

$$(26) \quad \|\Delta v_N\| = O(\|\Delta u_N\|) + O(\|\Delta u_B\|) + O(\|\Delta s\|) = O(\mu).$$

(24), (25), and (26) imply the desired results. $\square$

LEMMA 4.5. *If Assumptions 1, 2, and 3 hold, then there exists a constant c_2 independent of k such that for any $\theta \in (0, 1]$ and k large enough*

$$\|h(\theta)\|_\infty \leq c_2 \mu^2.$$

Proof. By Lemma 4.4 for k large enough we have $\|\Delta u\| \leq \delta$, where δ is the constant in Assumption 3. Hence by Assumption 3 we have for any $\theta \in (0, 1]$

$$|F_j(u + \theta \Delta u) - F_j(u) - \theta F'_j(u) \Delta u| \leq L \theta^2 \|\Delta u\|^2 \quad \forall j \in B.$$

According to the boundedness of u and Δu and Lemma 4.4, we have for all $j \in B$

$$\begin{aligned} |h_j(\theta)| &= \left| \theta^2 \Delta u_j \Delta v_j + (u_j + \theta \Delta u_j) [F_j(u + \theta \Delta u) - F_j(u) - \theta F'_j(u) \Delta u] \right| \\ &\leq \theta^2 |\Delta u_j \Delta v_j| + |u_j + \theta \Delta u_j| \cdot |F_j(u + \theta \Delta u) - F_j(u) - \theta F'_j(u) \Delta u| \\ &\leq \|\Delta u\| \cdot \|\Delta v\| + L \|u + \theta \Delta u\| \cdot \|\Delta u\|^2 \\ (27) \quad &= O(\mu^2). \end{aligned}$$

For $j \in N$, we use the boundedness of F' on $\mathcal{F}$ to estimate

$$\begin{aligned} &|F_j(u + \theta \Delta u) - F_j(u) - \theta F'_j(u) \Delta u| \\ &= \left| \int_0^\theta F'_j(u + t \Delta u) \Delta u dt - \theta F'_j(u) \Delta u \right| \\ &= O(\|\Delta u\|) = O(\mu). \end{aligned}$$

Furthermore, we have, by Lemmas 4.2 and 4.3, $|u_j| = O(\mu)$ and $|\Delta u_j| = O(\mu)$. Thus we have

$$\left| (u_j + \theta \Delta u_j)[F_j(u + \theta \Delta u) - F_j(u) - \theta F'_j(u) \Delta u] \right| = O(\mu^2) \quad \forall j \in N.$$

Then, similar to (27), we have for all $j \in N$

$$|h_j(\theta)| = O(\mu^2).$$

Summing up the arguments for $j \in B$ and $j \in N$, we complete the proof. $\square$

The main results concerning local convergence are as follows.

THEOREM 4.6. *Let*

$$\bar{\alpha} = \frac{(1 + \beta)c_2}{1 - \beta},$$

$\lambda \leq 1/4$, and $n \geq 2$. Let $\bar{k}$ be such that $0 < \mu_{\bar{k}} \leq \min\{1, \lambda/\bar{\alpha}\}$. Then the following statements hold true for all $k \geq \bar{k}$.

- (i) If $\lambda\mu_k \geq \zeta \geq \bar{\alpha}\mu_k^2$, then $(u(1), v(1)) \in \mathcal{N}_{\infty}^-(\beta)$ and $\mu(1) \leq (1 - 1/n)\mu_k$.
Consequently, we have $\theta_{k+1} = 1$.
- (ii) The algorithm can be in the linear mode at most once for all $k \geq \bar{k}$.
- (iii) $\{\mu_k\}$ converges to zero q -quadratically.

Proof. (i) Similar to the proofs of (11) and (12), we have

$$U(\theta)v(\theta) \geq (1 - \theta)Uv + \theta\zeta e - \|h(\theta)\|_{\infty}e$$

and

$$(28) \quad \mu(\theta) \leq (1 - \theta)\mu + \theta\zeta + \|h(\theta)\|_{\infty}.$$

Letting $\theta = 1$ in the above inequalities, we have

$$\begin{aligned} U(1)v(1) - \beta\mu(1)e &\geq (1 - \beta)\zeta e - (1 + \beta)\|h(1)\|_{\infty}e \\ &\geq (1 - \beta)\zeta e - (1 + \beta)c_2\mu_k^2e \\ &\geq 0. \end{aligned}$$

The second inequality above uses Lemma 4.5, and the last inequality is due to the condition $\zeta \geq \bar{\alpha}\mu_k^2$. This shows that $(u(1), v(1)) \in \mathcal{N}_{\infty}^-(\beta)$.

Setting $\theta = 1$ in (28), we obtain

$$(29) \quad \mu(1) \leq \zeta + \|h(1)\|_{\infty} \leq \lambda\mu_k + c_2\mu_k^2 \leq (1 - 1/n)\mu_k.$$

The last inequality follows from $c_2\mu_k \leq c_2\lambda/\bar{\alpha} \leq \lambda \leq 1 - \lambda - 1/n$. Then, according to Step 2 of the algorithm, we obtain $\theta_{k+1} = 1$.

(ii) Suppose that the algorithm is in the linear mode at some iteration $k \geq \bar{k}$. Then we have $\zeta = \lambda\mu_k$. According to (i), we have $\theta_{k+1} = 1$ and, of course, $\mu_{k+1} < 1$. Thus, at iteration $k + 1$, the algorithm switches to the quadratic mode with

$$(30) \quad \alpha = \lambda/\mu_{k+1} \geq \lambda/\mu_k \geq \bar{\alpha} \text{ and } \zeta = \alpha\mu_{k+1}^2 = \lambda\mu_{k+1}.$$

Hence we again have $\theta_{k+2} = 1$ according to (i). Consequently, iteration $k + 2$ will be in the quadratic mode with

$$(31) \quad \lambda\mu_{k+2} \geq \lambda\mu_{k+2}^2/\mu_{k+1} = \alpha\mu_{k+2}^2 = \zeta \geq \bar{\alpha}\mu_{k+2}^2.$$

The last inequality is due to (30). Now, by (i), we have $\theta_{k+3} = 1$ and iteration $k+3$ will be in the quadratic mode. Replacing $k+2$ with $k+i, i > 2$, in (31), we conclude that all subsequent iterations will be in the quadratic mode.

(iii) Since the algorithm is in quadratic mode for all sufficiently large k , we have $\zeta = \alpha\mu_k^2$ and $\theta_{k+1} = 1$ for all large k . Letting $\zeta = \alpha\mu_k^2$ in (29), we obtain

$$\mu_{k+1} = \mu(1) \leq \zeta + \|h(1)\|_\infty \leq \alpha\mu_k^2 + c_2\mu_k^2.$$

Thus $\{\mu_k\}$ converges to zero q -quadratically. $\square$

The following theorem is concerned with the r -quadratic convergence of the sequence $\{u^k, s^k, v^k\}_{k \geq \bar{k}}$.

THEOREM 4.7. *Suppose that $\{\mu_k\}$ converges quadratically, i.e., $\mu_{k+1} \leq c_3\mu_k^2$ for $k \geq \bar{k}$ and some $c_3 \geq 1$, and suppose that $c_3\mu_{\bar{k}} < 1$. Then the sequence $\{(u^k, s^k, v^k)\}_{k \geq \bar{k}}$ converges to a strictly complementary solution of (2) with r -order at least 2. That is, there exist constants $c_4 > 0$ and $\omega \in (0, 1)$ such that for all $t \geq 0$*

$$\|p^{\bar{k}+t} - p^*\| \leq c_4\omega^{2^t},$$

where $p = (u, s, v)$.

Proof. Let $\omega = c_3\mu_{\bar{k}}$ and $k = \bar{k} + t$. Then the quadratic convergence implies

$$\mu_k \leq c_3\mu_k \leq (c_3\mu_{k-1})^2 \leq \cdots \leq (c_3\mu_{\bar{k}})^{2^t} = \omega^{2^t}.$$

From Lemma 4.4 we have

$$\|u^{k+1} - u^k\| = \|\Delta u\| \leq c_1\mu_k \leq c_1\omega^{2^t}.$$

For any $i > k$

$$\begin{aligned} \|u^i - u^k\| &\leq \sum_{j=0}^{\infty} \|u^{k+j+1} - u^{k+j}\| \\ &\leq c_1 \sum_{j=0}^{\infty} \omega^{2^{t+j}} \leq \frac{c_1\omega^{2^t}}{1-\omega}. \end{aligned}$$

The estimate for s^k and v^k is similar. Hence the sequence $\{(u^k, s^k, v^k)\}$ is a Cauchy sequence; it must converge to a point (u^*, s^*, v^*) . The point (u^*, s^*, v^*) is a strict complementarity solution because of (20) and Theorem 4.6. Consequently, by letting $i \rightarrow \infty$,

$$\max \{\|u^* - u^k\|, \|s^* - s^k\|, \|v^* - v^k\|\} \leq \frac{c_1}{1-\omega} \omega^{2^t}.$$

This shows that $\{(u^k, s^k, v^k)\}_{k \geq \bar{k}}$ converges to (u^*, s^*, v^*) with an r -order of at least 2. $\square$

5. Final remarks. The main condition for global convergence of the proposed algorithm is the SLC. In a paper of Anderson and Ye [1] it is shown that, under such a condition, if a nonlinear monotone complementarity problem is polynomially solvable with an interior feasible starting point, then it can be polynomially solved without knowing such information at all by using a homogeneous self-dual reformulation of the problem. It is therefore a reasonable conjecture that some infeasible version of the algorithm will have similar convergence properties.

The major purpose of this paper is to provide a concise analysis on global and local convergence of the so-called largest-neighborhood long-step IP approach applied to nonlinear MVIPs. A novelty of the proposed algorithm is its self-adaptiveness in accelerating convergence, which, of course, is also applicable to solving linear problems.

In order to make our analysis streamlined, we have made a strong assumption (Assumption 2) in the analysis for local convergence. The assumption states that the Hessian matrix of the augmented Lagrangian with respect to the basic variables at each point of the set $\mathcal{F}$ is invertible. The set $\mathcal{F}$ is known to contain the sequence $\{u^k\}$ generated by the algorithm. Thus the assumption can be thought of as a global invertibility condition rather than a local requirement near the solution point u^* . It is interesting to note that this condition is different from the nondegeneracy conditions in the traditional sense. For instance, consider the following problem:

$$(32) \quad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad F(u) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad A = [1, -1], \quad \text{and } b = 0.$$

This problem is originated from the linear program

$$(33) \quad \begin{cases} \text{minimize} & u_1 + u_2 \\ \text{subject to} & u_1 - u_2 = 0, \quad u_1 \geq 0, \quad u_2 \geq 0. \end{cases}$$

At the unique solution $u^* = 0$ of (33) all three constraints of (33) are active, thus u^* is a degenerate solution in the sense of linear programming. However, Assumptions 1–3 are satisfied by (32) and the solution to (32) is

$$u^* = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad v^* = \begin{bmatrix} 1 - s^* \\ 1 + s^* \end{bmatrix} \quad \text{such that } -1 \leq s^* \leq 1.$$

Notice that v^* and s^* are not unique.

Other assumptions used in the literature for proving local quadratic convergence include

- B1 $F'(u)$ is positive definite;
- B2 $\|(UF'(\bar{u})U)^{-1}\| \leq M\|(UF'(u)U)^{-1}\| \quad \forall u, \bar{u} > 0$ and $\|U^{-1}(\bar{u} - u)\| \leq 1$; and
- B3 all solutions to (2) satisfy $u^* > 0$.

The conditions above are used in [19]. It can be seen that these assumptions are stronger than Assumptions 1, 2, and 3 and are not satisfied by example (32). Roughly speaking, Assumptions 1–3 are the “variational inequality version” of the “strongly nondegenerate” assumption used by Tseng in monotone complementarity problems [17]. An interesting problem for future research is how this assumption can be relaxed so as to make the algorithm applicable to a wide class of problems. A recent paper of Ralph and Wright [12] has made an attempt in this direction.

We would like to point out that by a suitable reformulation [12], problems with smooth convex constraints can be reduced to the form of problem (2). Therefore, the framework of analysis provided in our paper can be applied to problems with smooth convex constraints as well. Of course, since the reformulated function F will include gradients of the convex constraints, such an extension should be not trivial and should deserve a separate study.

Acknowledgment. We are grateful to Professor Y. Zhang and three anonymous referees whose comments helped to improve the manuscript.

REFERENCES

- [1] E. ANDERSON AND Y. YE, *On a homogeneous algorithm for the monotone complementarity problem*, Math. Programming, to appear.
- [2] D. DEN HERTOG, F. JARRE, C. ROOS, AND T. TERLAKY, *A sufficient condition for self-concordance, with application to some classes of structured convex programming problems*, Math. Programming, 69 (1995), pp. 75–88.
- [3] C. C. GONZAGA, *The largest step path following algorithm for monotone linear complementarity problems*, Math. Programming, 76 (1997), pp. 309–332.
- [4] O. GÜLER AND Y. YE, *Convergence behavior of interior point algorithms*, Math. Programming, 60 (1993), pp. 215–228.
- [5] B. JANSEN, K. ROOS, T. TERLAKY, AND A. YOSHISE, *Polynomiality of Primal-Dual Affine Scaling Algorithms for Nonlinear Complementarity Problems*, Discussion Paper Series No. 648, Institute of Socio-Economic Planning, University of Tsukuba, Japan, 1995.
- [6] M. KOJIMA, N. MEGIDDO, AND S. MIZUNO, *Theoretical convergence of large-step primal-dual interior point algorithms for linear programming*, Math. Programming, 59 (1992), pp. 1–21.
- [7] M. KOJIMA, S. MIZUNO, AND A. YOSHISE, *A primal-dual interior point method for linear programming*, in Progress in Mathematical Programming—Interior Point and Related Methods, N. Megiddo, ed., Springer, New York, 1989.
- [8] Z. Q. LUO AND Y. YE, *A genuine quadratically convergent polynomial interior point algorithm for linear programming*, in Advances in Optimization and Approximation, D. Du and J. Sun, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994, pp. 235–246.
- [9] S. MIZUNO, M. TODD, AND Y. YE, *On adaptive step primal-dual interior point algorithms for linear programming*, Math. Oper. Res., 18 (1993), pp. 964–981.
- [10] Y. NESTEROV AND A. NEMIROVSKII, *Interior Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, PA, 1994.
- [11] F. POTRA AND Y. YE, *Interior point methods for nonlinear complementarity problems*, J. Optim. Theory Appl., 88 (1996), pp. 617–647.
- [12] D. RALPH AND S. WRIGHT, *Superlinear Convergence of an Interior-Point Method for Monotone Variational Inequalities*, Preprint MCS-P556-0196, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1996.
- [13] J. SUN, K. WEE, AND J. ZHU, *An interior point method for solving a class of stochastic programming problems*, in Recent Advances in Nonsmooth Optimization, D. Du, L. Qi, and R. Womersley, eds., World Scientific Publishers, River Edge, NJ, 1995, pp. 392–404.
- [14] J. SUN AND J. ZHU, *A predictor-corrector algorithm for extended linear-quadratic programming*, Comput. Oper. Res., 23 (1996), pp. 755–767.
- [15] J. SUN, J. ZHU, AND G. ZHAO, *A predictor-corrector algorithm for a class of nonlinear saddle point problems*, SIAM J. Control Optim., 35 (1997), pp. 532–551.
- [16] P. TSENG, *Global linear convergence of a path-following algorithm for some monotone variational inequality problems*, J. Optim. Theory Appl., 75 (1992), pp. 265–279.
- [17] P. TSENG, *An infeasible path-following method for monotone complementarity problems*, SIAM J. Optim., 7 (1997), pp. 386–402.
- [18] S. J. WRIGHT AND D. RALPH, *A superlinear infeasible-interior-point algorithm for monotone complementarity problems*, Math. Oper. Res., 21 (1996), pp. 815–838.
- [19] J. H. WU, *Interior Point Algorithm for Some Monotone Variational Inequality Problems*, Preprint, Center for Research on Transportation, Université de Montréal, Montréal, QC, Canada, 1994.
- [20] Y. YE AND K. ANSTREICHER, *On quadratic and $O(\sqrt{n}L)$ convergence of a predictor-corrector algorithm for LCP*, Math. Programming, 62 (1993), pp. 537–552.
- [21] J. ZHU, *A path-following algorithm for a class of convex programming problems*, Z. Oper. Res., 36 (1992), pp. 359–377.

A TRUST REGION METHOD FOR SOLVING GENERALIZED COMPLEMENTARITY PROBLEMS*

HOUYUAN JIANG[†], MASAO FUKUSHIMA[‡], LIQUN QI[§], AND DEFENG SUN[§]

Abstract. Based on a semismooth equation reformulation using Fischer's function, a trust region algorithm is proposed for solving the generalized complementarity problem (GCP). The algorithm uses a generalized Jacobian of the function involved in the semismooth equation and adopts the squared natural residual of the semismooth equation as a merit function. The proposed algorithm is applicable to the nonlinear complementarity problem because the latter problem is a special case of the GCP. Global convergence and, under a nonsingularity assumption, local Q-superlinear (or quadratic) convergence of the algorithm are established. Moreover, calculation of a generalized Jacobian is discussed and numerical results are presented.

Key words. generalized complementarity problem, nonlinear complementarity problem, semismooth equation, trust region method, global and superlinear convergence

AMS subject classifications. 90C33, 65K10

PII. S1052623495296541

1. Introduction. The GCP is to find a vector $x \in \mathbb{R}^n$ such that

$$(1.1) \quad F(x) \geq 0, \quad G(x) \geq 0, \quad F(x)^T G(x) = 0,$$

where $F, G : \mathbb{R}^n \rightarrow \mathbb{R}^n$ are assumed to be continuously differentiable.

When $G(x) \equiv x - E(x)$, where $E : \mathbb{R}^n \rightarrow \mathbb{R}^n$, the GCP reduces to the so-called implicit complementarity problem [8, 28, 27]. In particular, if $G(x) \equiv x$, the GCP reduces to the nonlinear complementarity problem (NCP), which is a general framework for optimality conditions of mathematical programs as well as variational inequalities. Moreover, if F is an affine function, i.e., $F(x) = Mx + q$ with a matrix $M \in \mathbb{R}^{n \times n}$ and a vector $q \in \mathbb{R}^n$, then the NCP reduces to the linear complementarity problem (LCP), which in turn contains linear and quadratic programming problems as special cases.

There are many Newton-based methods for solving the NCP. We do not intend to give a short survey of numerical methods for solving the NCP. The interested reader is referred to two survey papers [18, 29] and to [1, 2, 15, 22, 30] for recent progress. In order to enlarge the domain of convergence of Newton-based methods, line search is usually used on certain merit functions derived from different reformulations of the NCP. A trust region strategy is used in [26] for solving a bound-constrained nonlinear least squares reformulation of the NCP. Global convergence of this trust region method was established, and, under the strict complementarity condition and other conditions, superlinear convergence was also given.

*Received by the editors December 21, 1995; accepted for publication (in revised form) December 17, 1996. The research of the first, third, and fourth authors was supported by the Australian Research Council; the research of the second author was supported in part by the Scientific Research Grant-in-Aid from the Ministry of Education, Science and Culture, Japan.

<http://www.siam.org/journals/siopt/8-1/29654.html>

[†]Department of Mathematics, The University of Melbourne, Parkville, Victoria 3052, Australia (jiang@mundoe.maths.mu.oz.au).

[‡]Department of Applied Mathematics and Physics, Graduate School of Engineering, Kyoto University, Kyoto 606-01, Japan (fuku@kuamp.kyoto-u.ac.jp).

[§]School of Mathematics, The University of New South Wales, Sydney 2052, Australia (qi@alpha.maths.unsw.edu.au, sun@alpha.maths.unsw.edu.au).

In this paper, we shall propose a trust region method for solving an unconstrained least squares reformulation of the GCP. It is surely applicable to the NCP since the GCP is a generalization of the NCP. The main difference between the proposed trust region method and the classical trust region methods lies in the updating rule for the trust region radius at the beginning of each iteration. More precisely, at the beginning of each iteration, the trust region radius is always set greater than a fixed (small) positive constant rather than solely updated from the final trust region radius of the last iteration as in the classical trust region methods. This type of updating rule for the trust region radius has been used in recent literature [14, 17, 23] for optimization problems and systems of nonsmooth equations. We will show that the proposed special strategy of updating trust region radii enables us to recover local superlinear convergence under some conditions in spite of the fact that the functions involved in the system of equations are only semismooth. (For globally and superlinearly convergent trust region methods for systems of smooth equations, see [25].)

The remainder of the paper is organized as follows. In the next section, the GCP will be converted into a system of semismooth equations and an unconstrained differentiable minimization problem. These two reformulations are equivalent to the GCP in a certain sense. In section 3, a trust region method is proposed for solving the GCP based on these two reformulations. In section 4, we discuss regularity conditions which ensure that a stationary point of the reformulated unconstrained minimization problem is a solution of the GCP. Section 5 is devoted to proving global convergence of the algorithm. In section 6, local superlinear convergence of the algorithm will be established under some nonsingularity condition. In section 7, we present a method for calculating a generalized Jacobian, which is needed in the algorithm. Numerical results are presented in section 8. We conclude our paper by giving some remarks in the final section.

A few words about our notation. The inner product of vectors $x, y \in \mathbb{R}^n$ is denoted by $x^T y$. Let $\|\cdot\|$, $\|\cdot\|_\infty$, and $\|\cdot\|_2$ denote any fixed norm, the l_∞ norm, and the Euclidean norm, respectively, in a space of appropriate dimension. The Jacobian of a vector function F at a point x is denoted by $\nabla F(x)^T$. If $M = (M_{ij})$ is an $n \times n$ matrix and α, β are two subsets of $\{1, \dots, n\}$, then $M_{\alpha\beta}$ denotes the submatrix of M consisting of elements M_{ij} with $i \in \alpha$ and $j \in \beta$.

2. Reformulation and preliminaries. Throughout this paper, we assume that F and G are continuously differentiable on $\mathbb{R}^n$.

Let $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}$ be defined by

$$(2.1) \quad \phi(a, b) := \sqrt{a^2 + b^2} - a - b.$$

A basic property of this function is that

$$\phi(a, b) = 0 \iff a \geq 0, b \geq 0, ab = 0.$$

From this property, the GCP (1.1) can readily be recast as the system of nonsmooth equations

$$(2.2) \quad H(x) = \begin{pmatrix} H_1(x) \\ \vdots \\ H_n(x) \end{pmatrix} := \begin{pmatrix} \phi(F_1(x), G_1(x)) \\ \vdots \\ \phi(F_n(x), G_n(x)) \end{pmatrix} = 0$$

in the sense that x solves (1.1) if and only if x solves (2.2). The function ϕ was first introduced by Fischer [11] (but attributed to Burmeister) and later used to study

various methods for solving the NCP and related problems; see survey paper [12] for details. Recently the function ϕ has been used to reformulate the GCP [20]. (See also [35] for other reformulations of the GCP.)

Note that H is locally Lipschitz on $\mathbb{R}^n$ and Fréchet differentiable on the set Ω , where

$$\Omega := \{x \in \mathbb{R}^n \mid F_i(x)^2 + G_i(x)^2 > 0, i = 1, \dots, n\}.$$

However, H is not necessarily differentiable at $x \notin \Omega$. Let $\partial H(x)$ denote the Clarke's generalized Jacobian of H at $x \in \mathbb{R}^n$ [3], which can be defined as the convex hull of the set $\partial_B H(x)$ [32], where

$$\partial_B H(x) = \left\{ V \in \mathbb{R}^{n \times n} \mid V = \lim_{x^k \rightarrow x} \nabla H(x^k)^T, H \text{ is differentiable at } x^k \text{ for all } k \right\}.$$

Similar to the discussions of [9, 19], we have for any $x \in \mathbb{R}^n$

$$(2.3) \quad \partial H(x) \subseteq \mathcal{D}^F(x) \nabla F(x)^T + \mathcal{D}^G(x) \nabla G(x)^T,$$

where $\mathcal{D}^F(x)$ and $\mathcal{D}^G(x)$ are sets of $n \times n$ diagonal matrices such that, for each pair $(D^F, D^G) = (\text{diag} \{D_1^F, \dots, D_n^F\}, \text{diag} \{D_1^G, \dots, D_n^G\}) \in \mathcal{D}^F(x) \times \mathcal{D}^G(x)$, the following conditions are satisfied:

$$(2.4) \quad (D_i^F + 1)^2 + (D_i^G + 1)^2 \leq 1, \quad i = 1, \dots, n.$$

In particular, if $F_i(x)^2 + G_i(x)^2 > 0$ for all $i = 1, \dots, n$, then we have $\mathcal{D}^F(x) = \{D^F\}$ and $\mathcal{D}^G(x) = \{D^G\}$ with

$$(2.5) \quad D_i^F = \frac{F_i(x)}{\sqrt{F_i(x)^2 + G_i(x)^2}} - 1, \quad D_i^G = \frac{G_i(x)}{\sqrt{F_i(x)^2 + G_i(x)^2}} - 1, \quad i = 1, \dots, n,$$

and $\nabla H(x)$ exists and is given by

$$\nabla H(x)^T = D^F \nabla F(x)^T + D^G \nabla G(x)^T.$$

Define a merit function $\Phi : \mathbb{R}^n \rightarrow \mathbb{R}$ as

$$\Phi(x) := \frac{1}{2} \sum_{i=1}^n \phi(F_i(x), G_i(x))^2 = \frac{1}{2} \|H(x)\|_2^2.$$

When the GCP has a solution, solving (1.1) is equivalent to finding a global minimum point of the unconstrained minimization problem

$$(2.6) \quad \min_{x \in \mathbb{R}^n} \Phi(x).$$

A favorable property of Φ is that it is continuously differentiable on the whole space $\mathbb{R}^n$, although H itself is not continuously differentiable in general [20]. It is easy to verify that for any $x \in \mathbb{R}^n$ and any $V \in \partial H(x)$

$$(2.7) \quad \nabla \Phi(x) = V^T H(x) = \nabla F(x) D^F H(x) + \nabla G(x) D^G H(x),$$

where D^F and D^G are diagonal matrices satisfying (2.4) such that $V = D^F \nabla F(x)^T + D^G \nabla G(x)^T$. The unconstrained minimization reformulation (2.6) as well as the semismooth equation reformulation (2.2) will be the basis of the proposed trust region method for solving the GCP.

We now introduce some definitions which will be useful in the subsequent analysis. Let $E : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be locally Lipschitz around $x \in \mathbb{R}^n$. E is said to be semismooth at x if

$$\lim_{\substack{V \in \partial E(x+th') \\ h' \rightarrow h, t \downarrow 0}} Vh'$$

exists for any $h \in \mathbb{R}^n$ [33]. It has been proved [33] that $E'(x; h)$, the directional derivative of E at x along the direction h , exists for any $h \in \mathbb{R}^n$ if E is semismooth at x . Moreover, E is said to be strongly semismooth at x if E is semismooth at x and for any $V \in \partial E(x + h)$, $h \rightarrow 0$,

$$Vh - E'(x; h) = O(\|h\|^2).$$

A matrix $M \in \mathbb{R}^{n \times n}$ is called a P -matrix if its principal minors are all positive, a P_0 -matrix if its principal minors are all nonnegative, and an S_0 -matrix if

$$\{x \in \mathbb{R}^n \mid x \geq 0, x \neq 0, Mx \geq 0\} \neq \emptyset.$$

It is known [4] that every P -matrix is a P_0 -matrix and every P_0 -matrix is an S_0 -matrix.

3. Algorithm. The proposed trust region algorithm for solving the GCP (1.1) is now formally stated.

ALGORITHM.

Step 0. Let $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \rho_1, \rho_2, \Delta_{\min}$, and Δ_1 be such that $0 < \alpha_1 < \alpha_2 < 1 < \alpha_3 < \alpha_4$, $0 < \rho_1 < \rho_2 < 1$, $\Delta_{\min} > 0$, and $\Delta_1 > 0$. Let $x^1 \in \mathbb{R}^n$ be a starting point. Set $k := 1$.

Step 1. If $\nabla \Phi(x^k) = 0$, stop. Otherwise, let $\hat{\Delta} := \max\{\Delta_{\min}, \Delta_k\}$ and choose $V_k \in \partial_B H(x^k)$.

Step 2. Let $\hat{s}$ be a solution of the minimization problem

$$(3.1) \quad \begin{aligned} & \min \nabla \Phi(x^k)^T s + \frac{1}{2} s^T V_k^T V_k s \\ & \text{s.t. } \|s\| \leq \hat{\Delta}. \end{aligned}$$

Step 3. Let

$$\hat{r} := \frac{\Phi(x^k + \hat{s}) - \Phi(x^k)}{\frac{1}{2} \|H(x^k) + V_k \hat{s}\|_2^2 - \Phi(x^k)}.$$

If $\hat{r} \geq \rho_1$, then let $s^k := \hat{s}$, $x^{k+1} := x^k + s^k$, $\delta_k := \hat{\Delta}$. For a fixed $\alpha \in [\alpha_3, \alpha_4]$, let

$$\Delta_{k+1} := \begin{cases} \hat{\Delta} & \text{if } \rho_1 \leq \hat{r} < \rho_2, \\ \alpha \hat{\Delta} & \text{if } \hat{r} \geq \rho_2. \end{cases}$$

Let k be replaced by $k + 1$, and return to Step 1. Otherwise, choose $\Delta \in [\alpha_1 \hat{\Delta}, \alpha_2 \hat{\Delta}]$. Let $\hat{\Delta} := \Delta$, and repeat Step 2.

Remarks. (i) In Step 2 of the algorithm, the minimization problem (3.1) may have several solutions. From the first equality in (2.7), it is easy to see that, for any $s \in \mathbb{R}^n$,

$$(3.2) \quad \frac{1}{2} \|H(x^k) + V_k s\|_2^2 - \Phi(x^k) = \nabla \Phi(x^k)^T s + \frac{1}{2} s^T V_k^T V_k s.$$

(ii) The main difference between the proposed trust region method and the classical trust region methods lies in the updating rule for the trust region radius at the beginning of each iteration. More precisely, at the beginning of each iteration k , the trust region radius is always set greater than the fixed positive constant $\Delta_{\min}$ rather than solely updated from the final trust region radius of iteration $k - 1$ as in the classical trust region methods. This type of updating rule for the trust region radius is also used in recent literature [14, 17, 23]. As we shall see, this special strategy enables us not only to establish global convergence but also to recover local superlinear convergence of the algorithm under some conditions, despite that (2.2) is a system of nonsmooth equations.

(iii) Note that δ_k does not play any role in the algorithm. But it will be useful in proving convergence theorems in the subsequent sections.

PROPOSITION 3.1. *The algorithm cannot cycle between Step 2 and Step 3 infinitely, provided that x^k is not a stationary point of Φ .*

Proof. Suppose the algorithm does cycle for some k and x^k is not a stationary point of Φ ; i.e., $\nabla\Phi(x^k) \neq 0$. Then the continuous differentiability of Φ implies

$$\begin{aligned}
 \Phi(x^k + \hat{s}) - \Phi(x^k) &= \nabla\Phi(x^k)^T \hat{s} + o(\|\hat{s}\|) \\
 &\leq \nabla\Phi(x^k)^T \hat{s} + \frac{1}{2} \hat{s}^T V_k^T V_k \hat{s} + o(\|\hat{s}\|) \\
 (3.3) \quad &= \frac{1}{2} \|H(x^k) + V_k \hat{s}\|_2^2 - \Phi(x^k) + o(\|\hat{s}\|),
 \end{aligned}$$

where the last equality follows from (3.2). Let $s^* = -\frac{\nabla\Phi(x^k)}{\|\nabla\Phi(x^k)\|}$. Clearly, $\tau\hat{\Delta}s^*$ is a feasible solution of (3.1) for any $\tau \in [0, 1]$. Since $\hat{s}$ solves (3.1), it follows from (3.2) that for $0 \leq \tau \leq 1$

$$\begin{aligned}
 \frac{1}{2} \|H(x^k) + V_k \hat{s}\|_2^2 - \Phi(x^k) &\leq \tau \hat{\Delta} \nabla\Phi(x^k)^T s^* + \frac{1}{2} \tau^2 \hat{\Delta}^2 (s^*)^T V_k^T V_k s^* \\
 &= -\tau \hat{\Delta} \|\nabla\Phi(x^k)\| + \frac{1}{2} \tau^2 \hat{\Delta}^2 (s^*)^T V_k^T V_k s^*.
 \end{aligned}$$

By the assumption that the algorithm cycles between Steps 2 and 3, both $\|\hat{s}\|$ and $\hat{\Delta}$ tend to zero, while $\hat{r} < \rho_1$ is maintained. Consequently, there exist $\tau^* > 0$ and $\Delta^* > 0$ such that when $\hat{\Delta} \leq \Delta^*$

$$\frac{1}{2} \|H(x^k) + V_k \hat{s}\|_2^2 - \Phi(x^k) \leq -\tau^* \hat{\Delta} \|\nabla\Phi(x^k)\|.$$

This, together with the definition of $\hat{r}$ and the relation (3.3), implies that

$$\hat{r} \geq 1 - \frac{o(\|\hat{s}\|)}{\tau^* \hat{\Delta} \|\nabla\Phi(x^k)\|} \geq 1 - \frac{o(\|\hat{\Delta}\|)}{\tau^* \hat{\Delta} \|\nabla\Phi(x^k)\|} \rightarrow 1 \quad \text{as } \hat{\Delta} \rightarrow 0,$$

which indicates that we eventually have $\hat{r} \geq \rho_1$. This is a contradiction and the desired result follows. $\square$

4. Regularity condition. Most unconstrained minimization methods normally generate a sequence converging to a local minimizer or a stationary point rather than a global minimizer. As discussed in section 2, if the GCP has a solution, then solving the GCP is equivalent to finding a global minimizer of the unconstrained minimization problem (2.6). It is therefore crucial to study under what conditions a stationary point

of (2.6) becomes a solution of the GCP. To this end, let us introduce some index sets: $\mathcal{C}(x)$, $\mathcal{R}(x)$, $\mathcal{P}(x)$, and $\mathcal{N}(x)$, which stand for complementarity indices, residual indices, positive indices, and negative indices, respectively; i.e.,

$$\begin{aligned}\mathcal{C}(x) &:= \{i \in \{1, \dots, n\} \mid F_i(x) \geq 0, G_i(x) \geq 0, F_i(x)G_i(x) = 0\}, \\ \mathcal{R}(x) &:= \{i \in \{1, \dots, n\} \mid i \notin \mathcal{C}(x)\}, \\ \mathcal{P}(x) &:= \{i \in \mathcal{R}(x) \mid F_i(x) > 0, G_i(x) > 0\}, \\ \mathcal{N}(x) &:= \mathcal{R}(x) \setminus \mathcal{P}(x).\end{aligned}$$

In the rest of the paper, we shall simply denote these sets as $\mathcal{C}$, $\mathcal{R}$, $\mathcal{P}$, and $\mathcal{N}$; the point x under consideration will always be clear from the context.

It is easy to verify the following result using (2.3), (2.4), and (2.5). The proof is omitted.

PROPOSITION 4.1. *Let diagonal matrices D^F and D^G satisfy (2.4) such that $D^F \nabla F(x)^T + D^G \nabla G(x)^T \in \partial H(x)$. Then for each $x \in \mathbb{R}^n$, we have the following relations:*

$$\begin{aligned}(D^F H(x))_i > 0 &\iff (D^G H(x))_i > 0 \iff i \in \mathcal{P}, \\ (D^F H(x))_i = 0 &\iff (D^G H(x))_i = 0 \iff i \in \mathcal{C}, \\ (D^F H(x))_i < 0 &\iff (D^G H(x))_i < 0 \iff i \in \mathcal{N}.\end{aligned}$$

Note that, since D^F and D^G are diagonal matrices, we have $(D^F H(x))_i = D_i^F H_i(x)$ and $(D^G H(x))_i = D_i^G H_i(x)$ for each $i = 1, \dots, n$.

DEFINITION 4.2. *The GCP is said to be regular at a point x if for any two vectors $z^1 \neq 0$, $z^2 \neq 0$ in $\mathbb{R}^n$ satisfying*

$$\begin{aligned}z_{\mathcal{C}}^1 &= 0, & z_{\mathcal{P}}^1 &> 0, & z_{\mathcal{N}}^1 &< 0, \\ z_{\mathcal{C}}^2 &= 0, & z_{\mathcal{P}}^2 &> 0, & z_{\mathcal{N}}^2 &< 0,\end{aligned}$$

we have

$$\nabla F(x)z^1 + \nabla G(x)z^2 \neq 0.$$

Moreover, a stationary point x of Φ is called a regular stationary point if the GCP is regular at x .

It is not hard to see that, when the GCP reduces to the NCP, the regularity condition introduced here is slightly weaker than that defined in [5]. Under the present regularity condition, the following theorem establishes the equivalence between solutions of the GCP and stationary points of the merit function Φ .

THEOREM 4.3. *x is a solution of the GCP if and only if x is a regular stationary point of Φ .*

Proof. If x is a solution of the GCP, then x is a stationary point of Φ , and, hence, $\mathcal{P} = \mathcal{N} = \emptyset$. By definition, x is a regular point of Φ .

Conversely, suppose x is a regular stationary point of Φ . Let D^F and D^G satisfy the conditions of Proposition 4.1. Let $z^1 = D^F H(x)$ and $z^2 = D^G H(x)$. Then

$$(4.1) \quad 0 = \nabla \Phi(x) = \nabla F(x)D^F H(x) + \nabla G(x)D^G H(x) = \nabla F(x)z^1 + \nabla G(x)z^2.$$

If x is not a solution of the GCP, it follows from Proposition 4.1 that $z^1 \neq 0$, $z^2 \neq 0$. By the definition of regularity and Proposition 4.1, for z^1 and z^2 ,

$$\nabla F(x)z^1 + \nabla G(x)z^2 \neq 0,$$

which contradicts (4.1). Therefore x is a solution of the GCP. $\square$

Next we present a sufficient condition for ensuring regularity of the GCP. The proof is based on a result of [5] for the NCP. In fact, when the GCP happens to be the NCP, the obtained result boils down to that of [5].

Let $D \in \mathbb{R}^{|\mathcal{R}| \times |\mathcal{R}|}$ denote a diagonal matrix with diagonal elements $D_1, \dots, D_{|\mathcal{R}|}$ defined by

$$D_i = \begin{cases} 1 & \text{if } i \in \mathcal{P}, \\ -1 & \text{if } i \in \mathcal{N}. \end{cases}$$

Evidently, DD is the $|\mathcal{R}| \times |\mathcal{R}|$ identity matrix. Using this notation, we establish the following proposition.

PROPOSITION 4.4. *Assume that $\nabla G(x)$ is invertible and*

$$D(\nabla F(x)^T(\nabla G(x)^T)^{-1})_{\mathcal{R}\mathcal{R}}D$$

is an S_0 -matrix. Then the GCP is regular at x .

Proof. Since $D(\nabla F(x)^T(\nabla G(x)^T)^{-1})_{\mathcal{R}\mathcal{R}}D$ is an S_0 -matrix, there exists a vector $y_{\mathcal{R}}^1 \neq 0$ such that

$$(4.2) \quad y_{\mathcal{R}}^1 \geq 0, \quad D(\nabla F(x)^T(\nabla G(x)^T)^{-1})_{\mathcal{R}\mathcal{R}}Dy_{\mathcal{R}}^1 \geq 0.$$

Let vectors $y^2 \in \mathbb{R}^n$ and $y \in \mathbb{R}^n$ be such that

$$(4.3) \quad y_{\mathcal{C}}^2 = 0, \quad y_{\mathcal{R}}^2 = Dy_{\mathcal{R}}^1,$$

$$(4.4) \quad y = (\nabla G(x)^{-1})^T y^2.$$

Clearly,

$$(4.5) \quad y_{\mathcal{R}}^2 \neq 0, \quad y_{\mathcal{P}}^2 \geq 0, \quad y_{\mathcal{N}}^2 \leq 0.$$

For any two vectors $z^1 \neq 0, z^2 \neq 0$ in $\mathbb{R}^n$ satisfying

$$\begin{aligned} z_{\mathcal{C}}^1 &= 0, & z_{\mathcal{P}}^1 &> 0, & z_{\mathcal{N}}^1 &< 0, \\ z_{\mathcal{C}}^2 &= 0, & z_{\mathcal{P}}^2 &> 0, & z_{\mathcal{N}}^2 &< 0, \end{aligned}$$

it follows from (4.3), (4.4), and (4.5) that

$$(4.6) \quad y^T \nabla G(x) z^2 = (y^2)^T z^2 = (y_{\mathcal{R}}^2)^T z_{\mathcal{R}}^2 > 0.$$

By the definition of y, y^2 , and z^1 , we have

$$\begin{aligned} y^T \nabla F(x) z^1 &= (y^2)^T \nabla G(x)^{-1} \nabla F(x) z^1 \\ &= (y_{\mathcal{R}}^2)^T (\nabla G(x)^{-1} \nabla F(x))_{\mathcal{R}\mathcal{R}} z_{\mathcal{R}}^1 \\ &= (y_{\mathcal{R}}^1)^T D(\nabla G(x)^{-1} \nabla F(x))_{\mathcal{R}\mathcal{R}} D D z_{\mathcal{R}}^1. \end{aligned}$$

Then (4.2) and the fact that $D z_{\mathcal{R}}^1 > 0$ imply that

$$y^T \nabla F(x) z^1 \geq 0,$$

which together with (4.6) yields

$$y^T (\nabla F(x) z^1 + \nabla G(x) z^2) > 0.$$

Consequently,

$$\nabla F(x)z^1 + \nabla G(x)z^2 \neq 0.$$

Therefore the GCP is regular at x . $\square$

It has been proved in [20] that any stationary point of Φ is a solution of the GCP if $\nabla G(x)^T$ is invertible and $\nabla F(x)^T(\nabla G(x)^T)^{-1}$ is a P_0 -matrix. It is not difficult to see that this result is a consequence of Proposition 4.4 by using the fact that a matrix M is a P_0 -matrix if and only if the matrix DMD is a P_0 -matrix for any nonsingular diagonal matrix D and the fact that any P_0 -matrix is an S_0 -matrix.

5. Global convergence. We now suppose that the algorithm generates an infinite sequence $\{x^k\}$; i.e., the stopping test in Step 1 of the algorithm is never fulfilled. Let $c > 0$ be a constant such that

$$\|y\| \leq c\|y\|_2$$

for all $y \in \mathbb{R}^n$. We first present a standard result.

LEMMA 5.1. *Let $\hat{s}$ be a solution of (3.1). Then*

$$\frac{1}{2}\|H(x^k) + V_k\hat{s}\|_2^2 - \Phi(x^k) \leq -\frac{1}{2}\|\nabla\Phi(x^k)\|_2 \min \left\{ \frac{\hat{\Delta}}{c}, \frac{\|\nabla\Phi(x^k)\|_2}{\|V_k^T V_k\|_2} \right\}.$$

Proof. Suppose $\tilde{s}$ is a solution of the following minimization problem

$$\begin{aligned} \min \quad & \nabla\Phi(x^k)^T s + \frac{1}{2}s^T V_k^T V_k s \\ \text{s.t.} \quad & \|s\|_2 \leq \frac{\hat{\Delta}}{c}. \end{aligned}$$

Then from Theorem 4 in [31],

$$(5.1) \quad \frac{1}{2}\|H(x^k) + V_k\tilde{s}\|_2^2 - \Phi(x^k) \leq -\frac{1}{2}\|\nabla\Phi(x^k)\|_2 \min \left\{ \frac{\hat{\Delta}}{c}, \frac{\|\nabla\Phi(x^k)\|_2}{\|V_k^T V_k\|_2} \right\}.$$

Since $\|\tilde{s}\| \leq c\|\tilde{s}\|_2 \leq \hat{\Delta}$, $\tilde{s}$ is a feasible solution of (3.1). Since $\hat{s}$ solves (3.1), the desired result follows from (3.2) and (5.1). $\square$

LEMMA 5.2. *Suppose x^* is the limit of a subsequence $\{x^k\}_{k \in K}$. If x^* is not a stationary point of Φ , then there exist $\hat{k}$ and $\Delta > 0$ such that for all $k \geq \hat{k}$ ($k \in K$)*

$$\hat{r} := \frac{\Phi(x^k + \hat{s}) - \Phi(x^k)}{\frac{1}{2}\|H(x^k) + V_k\hat{s}\|_2^2 - \Phi(x^k)} \geq \rho_1$$

whenever $\hat{\Delta} \in (0, \Delta)$, where $\hat{s}$ is a solution of (3.1).

Proof. First note that

$$(5.2) \quad \hat{r} = 1 + \frac{-\frac{1}{2}\hat{s}^T V_k^T V_k \hat{s} + o(\|\hat{s}\|_2)}{\frac{1}{2}\|H(x^k) + V_k\hat{s}\|_2^2 - \Phi(x^k)}.$$

Since x^* is not a stationary point and ∂H is upper semicontinuous, there exist positive constants β_1 and β_2 such that

$$(5.3) \quad \|\nabla\Phi(x^k)\|_2 \geq \beta_1, \quad \frac{\|\nabla\Phi(x^k)\|_2}{\|V_k^T V_k\|_2} \geq \beta_2$$

for all sufficiently large $k \in K$. By Lemma 5.1,

$$(5.4) \quad \frac{1}{2} \|H(x^k) + V_k \hat{s}\|_2^2 - \Phi(x^k) \leq -\frac{1}{2} \beta_1 \min \left\{ \frac{\hat{\Delta}}{c}, \beta_2 \right\}$$

for all sufficiently large $k \in K$. Then the desired result is a direct consequence of (5.2), (5.3), and (5.4). $\square$

LEMMA 5.3. *Suppose x^* is the limit of a subsequence $\{x^k\}_{k \in K}$. If x^* is not a stationary point of Φ , then*

$$\liminf_{k \in K, k \rightarrow \infty} \delta_k > 0,$$

where δ_k is defined as in the algorithm.

Proof. By Lemma 5.2, there exists a $\Delta > 0$ such that $\hat{r} \geq \rho_1$ whenever $\hat{\Delta} < \Delta$ at each iteration $k \in K$ sufficiently large. Thus, by the updating rule of the trust region radius in the algorithm, we have $\delta_k \geq \alpha_1 \Delta$ for all sufficiently large $k \in K$. The desired result follows. $\square$

Now we are ready to establish a global convergence theorem for the proposed trust region algorithm.

THEOREM 5.4. *Let $\{x^k\}$ be generated by the algorithm. Then any accumulation point of $\{x^k\}$ is a stationary point of Φ . Moreover, it is a solution of the GCP if the regularity condition holds at this point.*

Proof. Suppose x^* is an accumulation point of $\{x^k\}$, say $\lim_{k \in K, k \rightarrow \infty} x^k = x^*$. If x^* is not a stationary point of Φ , then Lemmas 5.1 and 5.3 imply that there exist $\beta_1 > 0$, $\beta_2 > 0$, $\delta > 0$, and $\hat{k}$ such that (5.4) holds and $\delta_k \geq \delta$ for all $k \geq \hat{k}$ ($k \in K$). By the algorithm, Lemma 5.1, (5.4), and the fact that Φ is nonnegative, we obtain

$$\begin{aligned} \Phi(x^1) &\geq \sum_{k=1}^{\infty} [\Phi(x^k) - \Phi(x^{k+1})] \\ &\geq \sum_{k=1}^{\infty} \rho_1 \left[\Phi(x^k) - \frac{1}{2} \|H(x^k) + V_k s^k\|_2^2 \right] \\ &\geq \sum_{\substack{k \geq \hat{k} \\ k \in K}} \rho_1 \left[\Phi(x^k) - \frac{1}{2} \|H(x^k) + V_k s^k\|_2^2 \right] \\ &\geq \rho_1 \sum_{\substack{k \geq \hat{k} \\ k \in K}} \frac{1}{2} \beta_1 \min \left\{ \frac{\delta_k}{c}, \beta_2 \right\} \\ &\geq \frac{1}{2} \rho_1 \beta_1 \sum_{\substack{k \geq \hat{k} \\ k \in K}} \min \left\{ \frac{\delta}{c}, \beta_2 \right\} = \infty. \end{aligned}$$

This is impossible. Therefore, x^* is a stationary point of Φ . $\square$

We now turn to the case where $\{x^k\}$ does not necessarily have an accumulation point.

THEOREM 5.5. *Let $\{x^k\}$ be generated by the algorithm. If $\{V_k\}$ is bounded, then $\{\nabla \Phi(x^k)\}$ is not bounded away from zero; that is,*

$$(5.5) \quad \liminf_{k \rightarrow \infty} \|\nabla \Phi(x^k)\| = 0.$$

Proof. Suppose (5.5) does not hold. Then the boundedness of $\{V_k\}$ and Lemma 5.1 imply the existence of β_1 and β_2 such that (5.4) holds for all k . By the algorithm and (5.4), we obtain

$$\begin{aligned}\Phi(x^1) &\geq \sum_{k=1}^{\infty} [\Phi(x^k) - \Phi(x^{k+1})] \\ &\geq \sum_{k=1}^{\infty} \rho_1 \left[\Phi(x^k) - \frac{1}{2} \|H(x^k) + V_k s^k\|_2^2 \right] \\ &\geq \frac{1}{2} \rho_1 \beta_1 \sum_{k=1}^{\infty} \min \left\{ \frac{\delta_k}{c}, \beta_2 \right\}.\end{aligned}$$

This implies

$$\sum_{k=1}^{\infty} \delta_k < \infty,$$

and, hence,

$$\sum_{k=1}^{\infty} \|x^{k+1} - x^k\| < \infty.$$

This implies that $\{x^k\}$ converges to a point x^* . By Theorem 5.4, x^* is a stationary point of Φ ; i.e., $\nabla \Phi(x^*) = 0$. This contradicts the assumption that

$$\liminf_{k \rightarrow \infty} \|\nabla \Phi(x^k)\| > 0. \quad \square$$

Remark. If ∇F and ∇G are bounded on $\mathbb{R}^n$, then ∂H is bounded on $\mathbb{R}^n$ and the boundedness assumption on $\{V_k\}$ is satisfied.

This theorem says that the sequence $\{x^k\}$ generated by the algorithm contains a stationary subsequence $\{x^k\}_{k \in K}$ in the sense that $\lim_{k \rightarrow \infty, k \in K} \|\nabla \Phi(x^k)\| = 0$, even if $\{x^k\}$ is unbounded. However, a stationary sequence is not necessarily a minimizing sequence of Φ in general. Conditions under which any stationary sequence of Φ is a minimizing sequence have been studied in [16] for the NCP.

6. Superlinear convergence. In this section, we shall be concerned with the rate of convergence of the algorithm. It is known [33, 32] that semismoothness and certain nonsingularity conditions at a solution of the system of nonsmooth equations play a crucial role in establishing superlinear convergence of some generalized Newton methods. Recall that H is said to be BD -regular at a point x if all the elements in $\partial_B H(x)$ are nonsingular [32]. We now derive some sufficient conditions for ensuring BD -regularity at a solution of the GCP.

Suppose $\nabla G(\bar{x})^T$ is invertible at a solution $\bar{x}$ of the GCP. Let

$$M = \begin{pmatrix} M_{\alpha\alpha} & M_{\alpha\beta} & M_{\alpha\gamma} \\ M_{\beta\alpha} & M_{\beta\beta} & M_{\beta\gamma} \\ M_{\gamma\alpha} & M_{\gamma\beta} & M_{\gamma\gamma} \end{pmatrix} := \nabla F(\bar{x})^T (\nabla G(\bar{x})^T)^{-1},$$

where

$$\begin{aligned}\alpha &:= \{i \in \{1, \dots, n\} \mid F_i(\bar{x}) = 0, G_i(\bar{x}) > 0\}, \\ \beta &:= \{i \in \{1, \dots, n\} \mid F_i(\bar{x}) = 0, G_i(\bar{x}) = 0\}, \\ \gamma &:= \{i \in \{1, \dots, n\} \mid F_i(\bar{x}) > 0, G_i(\bar{x}) = 0\}.\end{aligned}$$

The GCP is said to be R -regular at $\bar{x}$ if $M_{\alpha\alpha}$ is nonsingular and the Schur complement of $M_{\alpha\alpha}$ in the matrix

$$\begin{pmatrix} M_{\alpha\alpha} & M_{\alpha\beta} \\ M_{\beta\alpha} & M_{\beta\beta} \end{pmatrix};$$

i.e., $M_{\beta\beta} - M_{\beta\alpha}M_{\alpha\alpha}^{-1}M_{\alpha\beta}$ is a P -matrix, see [34].

PROPOSITION 6.1. *Let $\bar{x}$ be a solution of the GCP. Suppose $\nabla G(\bar{x})^T$ is invertible. If the GCP is R -regular at $\bar{x}$, then any $V \in \partial H(\bar{x})$ is nonsingular.*

Proof. Since $\nabla G(\bar{x})^T$ is invertible, by (2.3) any $V \in \partial H(\bar{x})$ can be represented as

$$\begin{aligned} V &= D^F \nabla F(\bar{x})^T + D^G \nabla G(\bar{x})^T \\ &= (D^F \nabla F(\bar{x})^T (\nabla G(\bar{x})^T)^{-1} + D^G) \nabla G(\bar{x})^T \\ &= (D^F M + D^G) \nabla G(\bar{x})^T, \end{aligned}$$

where D^F and D^G satisfy the condition (2.4) and $M = \nabla F(\bar{x})^T (\nabla G(\bar{x})^T)^{-1}$. Then V is nonsingular if and only if $D^F M + D^G$ is nonsingular. By some standard analysis (see, e.g., [9]), the nonsingularity of $D^F M + D^G$ can be deduced from the fact that the GCP is R -regular at x . $\square$

COROLLARY 6.2. *Suppose $\nabla G(\bar{x})^T$ is invertible. If $\nabla F(\bar{x})^T (\nabla G(\bar{x})^T)^{-1}$ is a P -matrix, then any $V \in \partial H(\bar{x})$ is nonsingular.*

Proof. Note that every principal submatrix of a P -matrix is a P -matrix, hence nonsingular, and the Schur complement of every principal submatrix of a P -matrix is a P -matrix. Therefore, the GCP is R -regular at $\bar{x}$. The desired result follows from Proposition 6.1. $\square$

LEMMA 6.3. (a) H is semismooth on $\mathbb{R}^n$ if F and G are continuously differentiable on $\mathbb{R}^n$; (b) H is strongly semismooth on $\mathbb{R}^n$ if ∇F and ∇G are Lipschitz continuous on $\mathbb{R}^n$.

Proof. Apparently, H is (strongly) semismooth at x if and only if each component of H is (strongly) semismooth at x . Therefore it suffices to prove that H_1 is (strongly) semismooth at any $x \in \mathbb{R}^n$ under different assumptions.

Note that H_1 can be regarded as a composition of the function $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}$ defined by (2.1) and the function $h : \mathbb{R}^n \rightarrow \mathbb{R}^2$ with $h(x) = (F_1(x), G_1(x))$; i.e., $H_1(x) = \phi(h(x))$. It is known that the composition of semismooth functions are semismooth [24, Theorem 5], and the composition of strongly semismooth functions are strongly semismooth [13, Theorem 5.7]. By Lemma 5.6 in [13], ϕ is strongly semismooth, hence semismooth on $\mathbb{R}^2$. Therefore (a) holds if F and G are semismooth on $\mathbb{R}^n$, and (b) holds if F and G are strongly semismooth on $\mathbb{R}^n$. On the other hand, the semismoothness of F and G follows from the continuous differentiability of F and G , and the strong semismoothness of F and G follows from the Lipschitz continuity of the Jacobians of F and G . $\square$

LEMMA 6.4. *Let $\{x^k\}$ be generated by the algorithm and x^* be an accumulation point of $\{x^k\}$. If the BD-regularity condition holds at x^* , then x^* is a solution of the GCP and there exists an open neighborhood $\mathcal{N}(x^*)$ of x^* such that when $x^k \in \mathcal{N}(x^*)$, we have*

$$x^{k+1} = x^k - V_k^{-1} H(x^k) \in \mathcal{N}(x^*).$$

Proof. By the BD-regularity condition and Theorem 5.4, x^* is a solution of the GCP and $V_k^{-1} H(x^k) \rightarrow 0$ as $x^k \rightarrow x^*$. Therefore, when x^k is sufficiently close to x^* ,

we have $\|V_k^{-1}H(x^k)\| \leq \Delta_{\min}$ and $-V_k^{-1}H(x^k)$ is a solution of (3.1) if $\hat{\Delta} \geq \Delta_{\min}$. On the other hand, we have $\|H(x^k - V_k^{-1}H(x^k))\| = o(\|H(x^k)\|)$ as $x^k \rightarrow x^*$ by Theorem 3.1 in [32]. This implies that $\hat{r} \geq \rho_1$ holds for some $\hat{\Delta} \geq \Delta_{\min}$ when x^k is sufficiently close to x^* . This in turn implies that, by the updating rule of the trust region radius, $\Delta_k \geq \Delta_{\min}$ for all k large enough. Therefore we have $x^{k+1} = x^k - V_k^{-1}H(x^k)$ for all k sufficiently large, and $\|x^{k+1} - x^*\| = o(\|x^k - x^*\|)$ as $x^k \rightarrow x^*$ from Theorem 3.1 in [32]. The conclusion follows easily. $\square$

We are now in a position to present rate of convergence results for the proposed algorithm.

THEOREM 6.5. *Suppose that the BD-regularity condition holds at an accumulation point x^* of the sequence $\{x^k\}$ generated by the algorithm. Then the entire sequence $\{x^k\}$ converges to x^* Q-superlinearly if F and G are continuously differentiable on $\mathbb{R}^n$. Moreover, the rate of convergence is Q-quadratic if ∇F and ∇G are Lipschitz continuous on $\mathbb{R}^n$.*

Proof. Since x^* is an accumulation point of $\{x^k\}$, there exists a subsequence $\{x^k\}_{k \in K}$ such that $\lim_{k \in K, k \rightarrow \infty} x^k = x^*$. By Lemma 6.4, there exists $\hat{k} \in K$ such that when $k \geq \hat{k}$

$$x^{k+1} = x^k - V_k^{-1}H(x^k).$$

It follows that the algorithm reduces to the generalized Newton method considered in [32]. Therefore, Q-superlinear convergence is guaranteed by Theorem 3.1 in [32]. Note that H is strongly semismooth if ∇F and ∇G are Lipschitz continuous. This implies Q-quadratic convergence again by Theorem 3.1 in [32]. $\square$

7. Computation of a generalized Jacobian. In the algorithm, we assumed that a generalized Jacobian of H is available at any point x . We now present a method for calculating a generalized Jacobian of H at x .

Define

$$I(x) := \{i \in \{1, \dots, n\} \mid F_i(x)^2 + G_i(x)^2 = 0, \|\nabla F_i(x)\| + \|\nabla G_i(x)\| > 0\}.$$

If $I(x) \neq \emptyset$, then we may assume, without loss of generality, that $I(x) = \{1, \dots, k\}$ for some $k \leq n$ and that $\nabla F_i(x) \neq 0$ for each $i \in I(x)$. Now we shall construct a vector $d \in \mathbb{R}^n$ such that

$$(7.1) \quad \nabla F_i(x)^T d \neq 0 \quad \text{for } i \in I(x).$$

First let $d := \nabla F_1(x)$ and

$$J(x) := \{i \in I(x) \mid \nabla F_i(x)^T d = 0\}.$$

If $J(x) = \emptyset$, then d satisfies (7.1). Otherwise, choose $j \in J(x)$, let

$$\bar{d} := \frac{\min_{i \in I(x) \setminus J(x)} |\nabla F_i(x)^T d|}{2 \max_{i \in J(x)} \|\nabla F_i(x)\| \|\nabla F_j(x)\|} \nabla F_j(x),$$

and put $\hat{d} := d + \bar{d}$. Then it is clear that $\nabla F_j(x)^T \hat{d} \neq 0$. Moreover, for $i \in I(x) \setminus J(x)$, we have

$$\nabla F_i(x)^T \hat{d} = \nabla F_i(x)^T d + \frac{\min_{i \in I(x) \setminus J(x)} |\nabla F_i(x)^T d|}{2 \max_{i \in J(x)} \|\nabla F_i(x)\| \|\nabla F_j(x)\|} \nabla F_i(x)^T \nabla F_j(x).$$

Since the absolute value of the second term on the right-hand side is never greater than that of the first term, we have $\nabla F_i(x)^T \hat{d} \neq 0$ for $i \in I(x) \setminus J(x)$. Define $\hat{J}(x)$ by

$$\hat{J}(x) := \{i \in I(x) \mid \nabla F_i(x)^T \hat{d} = 0\}.$$

The above arguments indicate that $\hat{J}(x) \subseteq J(x) \setminus \{j\}$. Put $d := \hat{d}$ and $J(x) := \hat{J}(x)$. If $J(x) = \emptyset$, then d satisfies (7.1). Otherwise, choose an index from $J(x)$ and repeat the above procedure. After at most k steps, we will have $J(x) = \emptyset$ and a vector d satisfying (7.1).

For small $t > 0$, let

$$y(t) := x + td + \hat{d}(t),$$

where $\|\hat{d}(t)\| = o(t)$ and d is a vector satisfying (7.1). Then H is differentiable at $y(t)$ with appropriately chosen $\hat{d}(t)$. Letting t tend to zero, we obtain a matrix $V := \lim_{t \rightarrow 0^+} \nabla H(y(t))^T$, which is an element of $\partial_B H(x)$ with the form $D^F \nabla F(x)^T + D^G \nabla G(x)^T$, where $D^F = \text{diag}\{D_1^F, \dots, D_n^F\}$ and $D^G = \text{diag}\{D_1^G, \dots, D_n^G\}$ are determined by

$$D_i^F = \begin{cases} \frac{F_i(x)}{\sqrt{F_i(x)^2 + G_i(x)^2}} - 1 & \text{if } F_i(x)^2 + G_i(x)^2 > 0, \\ \xi_i & \text{if } F_i(x) = G_i(x) = 0, \text{ and} \\ & \|\nabla F_i(x)\| + \|\nabla G_i(x)\| = 0, \\ \frac{\nabla F_i(x)^T d}{\sqrt{(\nabla F_i(x)^T d)^2 + (\nabla G_i(x)^T d)^2}} - 1 & \text{if } F_i(x) = G_i(x) = 0, \text{ and} \\ & \|\nabla F_i(x)\| + \|\nabla G_i(x)\| > 0, \end{cases}$$

$$D_i^G = \begin{cases} \frac{G_i(x)}{\sqrt{F_i(x)^2 + G_i(x)^2}} - 1 & \text{if } F_i(x)^2 + G_i(x)^2 > 0, \\ \eta_i & \text{if } F_i(x) = G_i(x) = 0, \text{ and} \\ & \|\nabla F_i(x)\| + \|\nabla G_i(x)\| = 0, \\ \frac{\nabla G_i(x)^T d}{\sqrt{(\nabla F_i(x)^T d)^2 + (\nabla G_i(x)^T d)^2}} - 1 & \text{if } F_i(x) = G_i(x) = 0, \text{ and} \\ & \|\nabla F_i(x)\| + \|\nabla G_i(x)\| > 0 \end{cases}$$

for $i = 1, \dots, n$, and ξ_i and η_i are some constants. Note that we have not specified any fixed numbers for D_i^F and D_i^G if $F_i(x) = G_i(x) = 0$ and $\|\nabla F_i(x)\| + \|\nabla G_i(x)\| = 0$. In practice, however, this does not cause any problem in calculating a generalized Jacobian $V \in \partial_B H(x)$. In fact, as shown easily, $\|\nabla F_i(x)\| + \|\nabla G_i(x)\| = 0$ implies that the i th row of any $V \in \partial_B H(x)$ becomes a zero vector.

8. Numerical experiments. In this section, we present some numerical experiments for the proposed algorithm. We have chosen the l_∞ -norm in the constraint set of the subproblem (3.1) so that (3.1) becomes a linear least squares problem with box constraints. We implemented a nonmonotone version of the algorithm in the sense that $\hat{r}$ in Step 3 of the algorithm is defined as

$$\hat{r} := \frac{\Phi(x^k + s) - \Psi_k}{\frac{1}{2} \|H(x^k) + V_k \hat{s}\|_2^2 - \Phi(x^k)},$$

where $\Psi_k := \max\{\Phi(x^l) \mid l = k - l_k, \dots, k\}$ and l_k is a nonnegative integer. This means that the objective function value is decreased compared with the maximum of the objective function values in the last $l_k + 1$ iterations, not necessarily decreased compared with the objective function value at the very last iteration. The nonmonotone version of the algorithm reduces to the algorithm in section 3 if $l_k \equiv 0$ for any k . In the code, we simply let $l_k = 3$ for $k \geq 4$ and $l_k = k - 1$ for $k = 2, 3$. The motivation to use the nonmonotone version of the algorithm is that the nonmonotone strategy can advance computational efficiency for complementarity problems, as observed in [6, 5, 21]. This has also been confirmed by our experience.

The algorithm was implemented in MATLAB and run on a SPARC 10 workstation. The subproblem (3.1) was solved by the **qp.m** inside MATLAB. Throughout the computational experiments, the parameters used in the algorithm were set as $\Delta_{\min} = 1.0$, $\Delta_1 = 100$, $\rho_1 = 10^{-4}$, $\rho_2 = 0.75$. The trust region radius is updated as follows: If $\hat{r} \geq \rho_2$, then $\Delta_{k+1} := 2\hat{\Delta}$; if $\hat{r} < \rho_1$, then $\hat{\Delta} := \hat{\Delta}/2$. We used

$$\min\{\|\min\{F(x), G(x)\}\|_\infty, \|\nabla\Phi(x)\|_2\} \leq 10^{-6}$$

as a stopping rule, where $\min\{F(x), G(x)\}$ denotes the vector with components $\min\{F_i(x), G_i(x)\}$, $i = 1, \dots, n$. Note that the second term on the left-hand side of the above stopping rule is used as a safeguard against the case that an accumulation point of the sequence generated by the algorithm is a mere stationary point of Φ , which is not a solution of the NCP or the GCP.

The code stated above was tested on the problems from libraries GAMSlib and MCPLIB [1, 7, 10]. For our purpose, we tested all linear and nonlinear complementarity problems from the libraries and leave other problems such as mixed complementarity problems in the sense of [1], nonlinear equations, and the KKT conditions of nonlinear programming problems for the future investigation when the corresponding theoretical results are justified. We have noted that some of the test problems such as **cammcp**, **hansmcp**, and **vonthmcp** in GAMSlib, and **choi** and **powell.mcp** in MCPLIB are actually not NCPs, although they were originally classified as NCPs [1]. Therefore **cammcp**, **hansmcp**, **vonthmcp**, **choi**, and **powell.mcp** were not tested for our code. On the other hand, we tested some problems such as those with the suffix “mge” in GAMSlib, and **colvdual**, **colvnlp**, **nash**, and **powell** in MCPLIB. These problems were not originally classified as NCPs.

The numerical results are summarized in Tables 8.1 and 8.2 for the problems from the libraries GAMSlib and MCPLIB, respectively. In Tables 8.1 and 8.2, **Dim** denotes the number of the variables in the problem, **Iter** denotes the number of iterations, which is also equal to the number of Jacobian evaluations for the function F and G , **NF** denotes the number of function evaluations for the functions F and G , and Φ denotes the final value of Φ at the found solution.

Two generalized complementarity problems in [27] were tested too. The interested reader is referred to [27] for full details of these two examples. The results are shown in Table 8.3. Three different starting points have been used, as distinguished by **Start** in Table 8.3.

Outrata-Zowe first problem [27]. Here $n = 4$. It is an implicit complementarity problem with both F and G being linear functions. We used the same three starting points as in [27].

Starting points: (a) $(0, 0, 0, 0)$, (b) $(-0.5, -0.5, -0.5, -0.5)$, (c) $(-1, -1, -1, -1)$.

Outrata-Zowe second problem [27]. Here $n = 4$. This problem is a modification of the last problem in which F is unchanged, but G is a nonlinear function.

TABLE 8.1
Numerical results for the problems from GAMS LIB.

Problem	Dim	Iter	NF	Φ
cafemge	101	31	95	5.1899e-13
cammge	128	0	1	5.09253e-13
co2mge	208	1	2	1.2755e-14
dmcmge	170	—	—	—
etamge	114	276	911	4.07511e-14
finmge	153	0	1	2.19764e-14
hansmge	43	13	23	3.46003e-23
harkmcp*	32	9	12	4.89955e-09
kehomge	9	11	19	8.14523e-20
mr5mcp	350	23	43	1.54433e-14
nsmge	212	21	42	1.4332e-16
oligomcp	6	6	7	5.41969e-17
sammge	23	0	1	0
scarfmcp	18	11	16	1.68098e-16
scarfmge	18	13	19	1.63513e-16
shovmge	51	1	2	5.58898e-14
threemge	9	0	1	0
transmcp	11	13	17	5.87216e-15
two3mcp	6	11	16	3.59176e-15
unstmge	5	8	19	1.56619e-16
vonthmge	80	—	—	—

TABLE 8.2
Numerical results for the problems from MCPLIB.

Problem	Dim	Iter	NF	Φ
bertsekas	15	17	42	1.97117e-16
billups	1	81	1085	4.97509e-05
colvdual	20	260	3458	1.08274e-4
colvnlp	15	27	61	5.79401e-14
cycle	1	3	10	1.70007e-21
explcp	16	19	42	3.6185e-14
hanskoop	14	20	54	4.56762e-18
josephy	4	26	45	2.97784e-14
kojshin	4	13	14	4.59541e-13
mathinum	3	4	5	4.33461e-17
mathisum	3	6	14	3.70779e-22
nash	10	8	9	9.61443e-20
pgvon105	105	18	38	1.25115e-13
pgvon106	106	—	—	—
powell	16	9	17	7.80373e-18
scarfanum	13	11	21	1.68104e-16
scarfasum	14	7	19	1.78446e-16
scarfbnum	39	25	47	5.58127e-15
scarfbsum	40	17	25	2.76111e-18
sppe	27	8	9	3.92819e-18
tobin	42	12	15	2.59193e-24

Starting points: (a) $(0, 0, 0, 0)$, (b) $(-0.5, -0.5, -0.5, -0.5)$, (c) $(-1, -1, -1, -1)$.

The numerical results presented in Tables 8.1, 8.2, and 8.3 show that the proposed method is viable for solving most NCPs from GAMS LIB and MCPLIB as well as the two GCPs efficiently. In Table 8.1, our code failed to solve **dmcmge** and **vonthmge** within 500 iterations. The problem **harkmcp** was solved by using some

TABLE 8.3
Numerical results for the two GCPs.

Problem	Dim	Start	Iter	NF	Φ
Outrata-Zowe 1	4	(a)	5	17	7.64995e-18
	4	(b)	4	16	9.7148e-15
	4	(c)	5	11	3.4347e-24
Outrata-Zowe 2	4	(a)	5	17	1.0474e-18
	4	(b)	4	16	4.88604e-15
	4	(c)	5	11	7.054e-22

minor modification of the code. Specifically, the Hessian of the objective function in the quadratic programming subproblem (3.1) was perturbed by adding $10^{-10} \times I$, where I is the identity matrix of an appropriate dimension, if the condition number of the Hessian is greater than 10^{15} . The reason we made this change in the code for **harmcqp** is that we noticed that the quadratic programming code **qp.m** failed to produce a solution when no perturbation was adopted. In Table 8.2, the code reached to a stationary point but it was an approximate solution point for the problem **colv-dual**. The code also failed to solve the problem **pgvon106** because the machine was unable to evaluate the objective function, i.e., $\Phi = \text{NaN}$, after the seventh iteration. However, we mention that **pgvon106** is not an NCP since some lower bounds of the variable x are 10^{-7} rather than zero and that the Jacobian of the function F is highly ill conditioned when components of x are close to zero. Notice in Table 8.1 that our code terminated without proceeding any iteration for some problems in GAMSLIB; i.e. Iter= 0. This is because the starting point provided in GAMSLIB is very close to the solution of the corresponding problem, which can also be observed from the value of Φ in the last column of Table 8.1.

9. Conclusions. In this paper, we have proposed a trust region method for solving the generalized complementarity problem by using both semismooth equation and differentiable optimization reformulations. The special trust region updating rule enables us to establish not only global convergence but also local superlinear convergence of the algorithm under some suitable conditions. We remark again that our trust region method is very different from other existing methods of using line search schemes as far as the globalization strategy is concerned. The proposed algorithm was implemented in MATLAB and was tested for all the NCPs from GAMSLIB and MCPLIB libraries. The preliminary numerical results presented show the viability of this method. The code successfully solved most of the test problems in a reasonably small number of function and Jacobian evaluations, although it failed or converged slowly in some cases. The latter fact suggests that there remain more issues to be addressed. This may be regarded as a further research topic. As expressed, we only tested NCPs from these two libraries. Therefore a possible future topic is to extend the proposed method to the mixed nonlinear complementarity problem which contains the variational inequality problem as a special case if the KKT reformulation of the latter problem is used.

Acknowledgments. The authors are grateful to Dr. Kouichi Taji for his helpful comment on Definition 4.2. We are thankful to Professor Michael Ferris for his kind help regarding the GAMSLIB and MCPLIB libraries. We are indebted to Professor Olvi Mangasarian and anonymous referees for their constructive comments which improved our numerical experiments. We also thank the Department of Electrical and Electronic Engineering at the University of Melbourne for providing access to the

SPARC 10 workstation used in numerical tests.

REFERENCES

- [1] S. C. BILLUPS, S. P. DIRKSE, AND M. C. FERRIS, *A comparison of large scale mixed complementarity problem solvers*, Comput. Optim. Appl., 7 (1997), pp. 3–25.
- [2] C. CHEN AND O. L. MANGASARIAN, *A class of smoothing functions for nonlinear and mixed complementarity problems*, Comput. Optim. Appl., 5 (1996), pp. 97–138.
- [3] F. H. CLARKE, *Optimization and Nonsmooth Analysis*, Wiley, New York, 1983.
- [4] R. W. COTTLE, J.-S. PANG, AND R. E. STONE, *The Linear Complementarity Problem*, Academic Press, New York, 1992.
- [5] T. DE LUCA, F. FACCHINEI, AND C. KANZOW, *A semismooth equation approach to the solution of nonlinear complementarity problems*, Math. Programming, 75 (1996), pp. 407–439.
- [6] S. P. DIRKSE AND M. C. FERRIS, *The PATH solver: A non-monotone stabilization scheme for mixed complementarity problems*, Optimization Methods and Software, 5 (1995), pp. 123–156.
- [7] S. P. DIRKSE AND M. C. FERRIS, *MCPLIB: A collection of nonlinear mixed complementarity problems*, Optimization Methods and Software, 5 (1995), pp. 319–345.
- [8] I. C. DOLCETTA AND U. MOSCO, *Implicit complementarity problems and quasi-variational inequalities*, in Variational Inequalities and Complementarity Problems: Theory and Applications, R. W. Cottle, F. Giannessi, and J.-L. Lions, eds., Wiley, New York, 1980, pp. 75–87.
- [9] F. FACCHINEI AND J. SOARES, *A new merit function for nonlinear complementarity problems and a related algorithm*, SIAM J. Optim., 7 (1997), pp. 225–247.
- [10] M. C. FERRIS AND T. F. RUTHERFORD, *Accessing realistic mixed complementarity problems within MATLAB*, in Nonlinear Optimization and Applications, G. Di Pillo and F. Giannessi, eds., Plenum Press, New York, 1996, pp. 141–153.
- [11] A. FISCHER, *A special Newton-type optimization method*, Optimization, 24 (1992), pp. 269–284.
- [12] A. FISCHER, *An NCP-function and its use for the solution of complementarity problems*, in Recent Advances in Nonsmooth Optimization, D. Du, L. Qi, and R. Womersley, eds., World Scientific Publishers, River Edge, NJ, 1995, pp. 88–105.
- [13] A. FISCHER, *Solution of monotone complementarity problems with locally Lipschitzian functions*, Math. Programming, 76 (1997), pp. 513–532.
- [14] A. FRIEDLANDER, J. M. MARTÍNEZ, AND A. SANTOS, *A new algorithm for bound constrained minimization*, J. Appl. Math. Optim., 30 (1994), pp. 235–266.
- [15] M. FUKUSHIMA, *Merit functions for variational inequality and complementarity problems*, in Nonlinear Optimization and Applications, G. Di Pillo and F. Giannessi, eds., Plenum Press, New York, 1996, pp. 155–170.
- [16] M. FUKUSHIMA AND J.-S. PANG, *Minimizing and stationary sequences of merit functions for complementarity problems and variational inequalities*, in Complementarity and Variational Problems: State of the Art, M. C. Ferris and J.-S. Pang, eds., SIAM, Philadelphia, PA, 1997, pp. 91–104.
- [17] S. A. GABRIEL AND J.-S. PANG, *A trust region method for constrained nonsmooth equations*, in Large Scale Optimization: State of the Art, W. W. Hager, D. W. Hearn, and P. M. Pardalos, eds., Kluwer Academic Publishers, Boston, MA, 1994, pp. 159–186.
- [18] P. T. HARKER AND J.-S. PANG, *Finite-dimensional variational inequality and nonlinear complementarity problem: A survey of theory, algorithms and applications*, Math. Programming, 48 (1990), pp. 161–220.
- [19] H. JIANG AND L. QI, *A new nonsmooth equations approach to nonlinear complementarity problems*, SIAM J. Control Optim., 35 (1997), pp. 178–193.
- [20] C. KANZOW AND M. FUKUSHIMA, *Equivalence of the generalized complementarity problem to differentiable unconstrained minimization*, J. Optim. Theory Appl., 90 (1996), pp. 581–603.
- [21] C. KANZOW AND M. FUKUSHIMA, *Theoretical and numerical investigation of the D-gap function for box constrained variational inequalities*, Math. Programming, to appear.
- [22] Z.-Q. LUO AND P. TSENG, *A new class of merit functions for the nonlinear complementarity problem*, in Complementarity and Variational Problems: State of the Art, M. C. Ferris and J.-S. Pang, eds., SIAM, Philadelphia, PA, 1997, pp. 204–225.
- [23] J. M. MARTÍNEZ AND A. SANTOS, *A trust region strategy for minimization on arbitrary domains*, Math. Programming, 68 (1995), pp. 267–302.
- [24] R. MIFFLIN, *Semismooth and semiconvex functions in constrained optimization*, SIAM J. Control Optim., 15 (1977), pp. 959–972.

- [25] J. J. MORÉ, *Recent developments in algorithms and software for trust region methods*, in Mathematical Programming: The State of the Art, A. Bachem, M. Grötschel, and B. Korte, eds., Springer-Verlag, Berlin, 1983, pp. 258–287.
- [26] J. J. MORÉ, *Global methods for nonlinear complementarity problems*, Math. Oper. Res., 21 (1996), pp. 589–614.
- [27] J. V. OUTRATA AND J. ZOWE, *A Newton method for a class of quasi-variational inequalities*, Comput. Optim. Appl., 4 (1995), pp. 5–21.
- [28] J.-S. PANG, *The implicit complementarity problem*, in Nonlinear Programming 4, O. L. Mangasarian, S. M. Robinson, and R. R. Meyer, eds., Academic Press, New York, 1981, pp. 487–518.
- [29] J.-S. PANG, *Complementarity problems*, in Handbook of Global Optimization, R. Horst and P. Pardalos, eds., Kluwer Academic Publishers, Boston, MA, 1995, pp. 271–338.
- [30] J.-S. PANG AND S. A. GABRIEL, *NE/SQP: A robust algorithm for the nonlinear complementarity problem*, Math. Programming, 60 (1993), pp. 295–337.
- [31] M. J. D. POWELL, *Convergence properties of a class of minimization algorithms*, in Nonlinear Programming 2, O. L. Mangasarian, R. R. Meyer, and S. M. Robinson, eds., Academic Press, New York, 1975, pp. 1–27.
- [32] L. QI, *Convergence analysis of some algorithms for solving nonsmooth equations*, Math. Oper. Res., 18 (1993), pp. 227–244.
- [33] L. QI AND J. SUN, *A nonsmooth version of Newton's method*, Math. Programming, 58 (1993), pp. 353–368.
- [34] S. M. ROBINSON, *Generalized equations*, in Mathematical Programming: The State of the Art, A. Bachem, M. Grötschel, and B. Korte, eds., Springer-Verlag, Berlin, 1983, pp. 346–367.
- [35] P. TSENG, N. YAMASHITA, AND M. FUKUSHIMA, *Equivalence of complementarity problems to differentiable minimization: A unified approach*, SIAM J. Optim., 6 (1996), pp. 446–460.

AN ACTIVE SET NEWTON ALGORITHM FOR LARGE-SCALE NONLINEAR PROGRAMS WITH BOX CONSTRAINTS*

FRANCISCO FACCHINEI[†], JOAQUIM JÚDICE[‡], AND JOÃO SOARES[§]

Abstract. A new algorithm for large-scale nonlinear programs with box constraints is introduced. The algorithm is based on an efficient identification technique of the active set at the solution and on a nonmonotone stabilization technique. It possesses global and superlinear convergence properties under standard assumptions. A new technique for generating test problems with known characteristics is also introduced. The implementation of the method is described along with computational results for large-scale problems.

Key words. Newton's method, box constraints, large-scale problems, active set

AMS subject classifications. 90C30, 65K05

PII. S1052623493253991

1. Introduction. In this paper we consider the solution of the box constrained nonlinear programming problem

$$(1.1) \quad \min_{x \in K} f(x),$$

where

$$(1.2) \quad K = \{x \in \mathbb{R}^n : l_i \leq x_i \leq u_i, i = 1, \dots, n\}$$

is a nonempty set. We assume that lower and upper bounds may be finite or infinite and that f is a twice continuously differentiable function in an open set containing K .

A vector $\bar{x} \in K$ is said to be a stationary point for Problem (1.1) if, for every $i = 1, \dots, n$,

$$(1.3) \quad \begin{cases} l_i = \bar{x}_i & \implies \nabla f_i(\bar{x}) \geq 0, \\ l_i < \bar{x}_i < u_i & \implies \nabla f_i(\bar{x}) = 0, \\ \bar{x}_i = u_i & \implies \nabla f_i(\bar{x}) \leq 0, \end{cases}$$

where $\nabla f_i(x)$ is the i th component of the gradient vector of f at x . Strict complementarity is said to hold at $\bar{x}$ if $\nabla f_i(\bar{x}) > 0$ and $\nabla f_i(\bar{x}) < 0$ in the first and third implication of (1.3).

Algorithms for solving Problem (1.1) fall mostly in the *active set* category. In this class of methods a *working set* estimates the set of active constraints at the solution and it is updated from iteration to iteration. In general, only a single active constraint can be added to or dropped from the working set at each iteration, and this can slow down the convergence rate, especially when dealing with large-scale problems.

*Received by the editors August 19, 1993; accepted for publication (in revised form) October 14, 1996.

<http://www.siam.org/journals/siopt/8-1/25399.html>

[†]Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza," Via Buonarroti 12, 00185 Roma, Italy (soler@dis.uniroma1.it).

[‡]Departamento de Matemática, Universidade de Coimbra, 3000 Coimbra, Portugal (judice@it.uc.pt).

[§]Columbia University, 804 Uris Hall, New York, NY 10027 (jsoares@corc.ieor.columbia.edu).

However, for the special case of Problem (1.1), it is possible to envisage algorithms that update the working set more efficiently [27], especially in the quadratic case [15].

In recent years, a number of algorithms have been proposed that attempt to quickly identify the correct active set. With regard to Problem (1.1), the seminal works are [2, 4] (see also [3]), where it is shown that, if the strict complementarity assumption holds, then it is possible, using a *projection* method, to add to or drop from the current estimated active set many constraints at each iteration and yet find the active set in a finite number of steps; this work has motivated further studies on projection techniques, both for the general linearly constrained case and for the box constrained case (see, e.g., [6, 8, 10, 22]). These algorithms are surely among the most efficient ones for the solution of large scale, convex, quadratic problems, [31, 32], but there is not sufficient numerical evidence of their effectiveness in the nonquadratic case.

More recently *trust region*-type algorithms for unconstrained optimization have been successfully extended to handle the presence of bounds on the variables. The global convergence theory thus developed is robust [12], and, under appropriate assumptions, it is possible to establish a superlinear convergence rate without requiring strict complementarity [30]. Furthermore, preliminary numerical results on small, dense problems [13] suggest that these methods may be well suited for large-scale problems.

We finally mention that interior point methods for the solution of Problem (1.1) are currently an active field of research. Some interesting theoretical results can be obtained in this framework, but computational experience with this class of methods for nonquadratic problems is still limited (see, however, [33]).

In this paper we propose and preliminarily test a new algorithm for the solution of Problem (1.1) that combines ideas from active set and nonmonotone Newton-type methods. It generates feasible iterates and has the general form $x^{k+1} = x^k + \alpha^k d^k$, where d^k is the *search direction* and α^k is the *stepsize*. To compute the direction d^k , a guessing technique is first employed to predict which bounds are active at the solution. This procedure, which permits fast changes in the working set, fixes some of the components of d^k . Then the components of d^k corresponding to the free variables are determined by solving a strictly convex, quadratic subproblem. This subproblem consists of minimizing the classical second-order Newton's model of the objective function in the subspace of free variables, subject to box constraints which guarantee that x^{k+1} is feasible. The stepsize α^k is computed by an extension of the *nonmonotone stabilization technique* proposed in [28]. This is a very flexible technique that has proven to be efficient in unconstrained optimization [28].

We show that the algorithm possesses global and quadratic convergence properties under standard assumptions. Furthermore, it reduces to the unconstrained minimization algorithm proposed in [28] when all the variables have no bounds.

From the computational point of view we note that the method requires the solution of a (reduced) box constrained quadratic subproblem at each iteration. One of the aims of this paper is to suggest that a careful implementation of this algorithm, using sparse matrix techniques, is suitable for the solution of large-scale box constrained programs. Direct methods for large-scale, box constrained optimization have been advocated as a viable approach for the quadratic case [10, 29], but very few results are reported in the literature for the nonquadratic case. To our knowledge LANCELOT [14] is the only package that provides, in a trust region framework, direct methods to solve large-scale, box constrained problems.

In this paper we also introduce a simple technique for generating box constrained test problems with a known minimum point. Among other things, the technique allows control over the number of active constraints at the solution and the degree of degeneracy. Furthermore, the Hessian of the objective function at the solution is the same as that of the unconstrained problem, so that particular features of the unconstrained problems (ill conditioning, singularity) can be preserved.

The paper is organized as follows. In the next section we introduce a general line search scheme for the solution of Problem (1.1). We show that global convergence can be ensured under appropriate assumptions on the search direction. In section 3 we describe a procedure to calculate the search direction in such a way that those assumptions are satisfied. The local convergence properties of the resulting algorithm are analyzed in section 4. In section 5 we examine some linear algebra issues related with an implementation of the algorithm for large-scale problems. In section 6 the test problem generator is described, while preliminary numerical results are reported in section 7. Finally, in section 8 some conclusions are drawn and some directions for future research are outlined.

Regarding the notation, if M is an $n \times n$ matrix with elements M_{ij} , $i, j = 1, \dots, n$, and I is an index set such that $I \subseteq \{1, \dots, n\}$, we denote by M_I the $|I| \times |I|$ submatrix of M consisting of elements M_{ij} , $i \in I, j \in I$. If w is an n vector, we denote by w_I the subvector with components w_i , $i \in I$. A superscript (k or j in general) indicates iteration numbers. Finally, $\|\cdot\|$ indicates the Euclidean norm or the corresponding subordinate matrix norm.

2. A framework for the solution of box constrained optimization problems. In this section we introduce a general scheme for the solution of box constrained problems that can be analyzed using tools similar to those employed in unconstrained optimization. This iterative scheme is a line search scheme with the following general structure.

GENERAL LINE SEARCH ALGORITHM.

Data: Choose $x^0 \in K$.

Initialization: Set $k = 0$.

Iteration-k: Compute a direction d^k .
If $d^k = 0$ STOP: x^k is a stationary point. Otherwise,
compute a stepsize $\alpha^k \in (0, 1]$ and set $x^{k+1} = x^k + \alpha^k d^k$, $k := k + 1$.

Note that since K is defined by box constraints, an initial point $x^0 \in K$ is readily available.

Next, we introduce two hypotheses which will be used in proving the properties of the algorithm.

HYPOTHESIS 1. *The level set $\mathcal{L}^0 = \{x \in \mathbb{R}^n: f(x) \leq f(x^0)\} \cap K$ is compact.*

HYPOTHESIS 2. *There exist scalars $\gamma > 0$ and $p > 1$ such that, for every $k = 0, 1, \dots$, $d^k \in \mathbb{R}^n$ satisfies the following conditions:*

- (a) $x^k + d^k \in K$;
- (b) $\nabla f(x^k)^T d^k \leq -\gamma \|d^k\|^p$;
- (c) $d^k = 0$ if and only if x^k is a stationary point of Problem (1.1);
- (d) if $x^k \rightarrow \bar{x}$ and $d^k \rightarrow 0$, then $\bar{x}$ is a stationary point of Problem (1.1).

If every variable has finite upper and lower bounds, or if the objective function is strongly convex, then Hypothesis 1 is trivially satisfied. This assumption will be used to guarantee that the algorithm generates a bounded sequence so that at least one

accumulation point exists. In the unconstrained case ($K = \mathbb{R}^n$) Hypothesis 1 reduces to the classical compactness assumption on the level sets of the objective function.

Hypothesis 2(a) and the convexity of K guarantee that $x^k \in K$ for every k . Hypothesis 2(b), (c), and (d) will be employed to guarantee that every limit point of the sequence $\{x^k\}$ is a stationary point of Problem (1.1). In the unconstrained case Hypothesis 2(b), (c), and (d) can take the place of more standard *gradient related* conditions [34, 3]. In the next section we indicate a possible way to define a search direction satisfying Hypothesis 2.

There are many ways to compute a suitable stepsize α^k . Among the most common procedures, the Armijo stabilization technique is probably the simplest one. We present it below.

ARMIJO STABILIZATION TECHNIQUE.

Data: Choose $x^0 \in K$, $\sigma \in (0, 1/2)$.

Initialization: Set $k = 0$.

Iteration-k: Compute a direction d^k ;
If $d^k = 0$ STOP. Otherwise,
find the smallest integer $i = 0, 1, \dots$ such that

$$(2.1) \quad f(x^k + 2^{-i}d^k) \leq f(x^k) + \sigma 2^{-i} \nabla f(x^k)^T d^k$$

and set $\alpha^k = 2^{-i}$.

Set $x^{k+1} = x^k + \alpha^k d^k$, $k := k + 1$.

The stopping criterion $d^k = 0$ is based on Hypothesis 2(c), while Hypothesis 2(b) guarantees that a positive α^k satisfying (2.1) can be found in a finite number of steps.

We now show that the Armijo scheme described above produces a sequence such that each of its limit points is a stationary point of Problem (1.1). This proof is instructive, since it illustrates well and simply how Hypothesis 2 can be used to adapt the classical proof techniques of unconstrained optimization to the box constrained case.

THEOREM 2.1. *If*

1. *Hypotheses 1 and 2 hold;*
2. *α^k is computed by the Armijo stabilization technique,*

then the sequence $\{x^k\}$ has at least a limit point and every limit point of this sequence is a stationary point for Problem (1.1).

Proof. It easily follows from Hypotheses 2(a) and (b) and the condition (2.1) that the sequence $\{x^k\}$ of the points generated by the algorithm is contained in the compact set $\mathcal{L}^0$. Hence, by Hypothesis 1, there exists at least a limit point of this sequence.

If the sequence $\{x^k\}$ is finite with last point $\bar{x}$ then, by Hypothesis 2(c), $\bar{x}$ is a stationary point of Problem (1.1). So assume that the sequence is infinite. Let L be the Lipschitz constant of ∇f over a convex open bounded set containing $\mathcal{L}^0$. Then we have, for every feasible x^k , direction $d^k \in \mathbb{R}^n$ and step length $\alpha^k \geq 0$,

$$f(x^k + \alpha^k d^k) - f(x^k) \leq \alpha^k \nabla f(x^k)^T d^k + \alpha^{k2} L \|d^k\|^2 / 2.$$

This inequality readily implies, together with the rule adopted in the Armijo stabilization technique for the computation of α^k , that $\alpha^k \geq (\sigma - 1) \nabla f(x^k)^T d^k / (L \|d^k\|^2)$,

so that by Hypothesis 2(b) we can write

$$\begin{aligned}
 f(x^k + \alpha^k d^k) - f(x^k) &\leq \sigma \alpha^k \nabla f(x^k)^T d^k \\
 &\leq -\sigma(\sigma - 1)(\nabla f(x^k)^T d^k)^2 / (L \|d^k\|^2) \\
 (2.2) \quad &\leq -\sigma \gamma^2 \|d^k\|^{2(p-1)} / L.
 \end{aligned}$$

Since the sequence $\{f(x^k)\}$ is decreasing and bounded from below, it follows from (2.2) and $p > 1$ that $\{d^k\} \rightarrow 0$ so that the theorem follows by Hypothesis 2(d). $\square$

It should be clear from Hypothesis 2 that any stabilization technique capable of driving $\nabla f(x^k)^T d^k$ to 0 could be used instead of the Armijo stabilization technique. It is then possible to show that almost any classical method for unconstrained optimization (which generally guarantees the condition $\nabla f(x^k)^T d^k \rightarrow 0$) can be extended to box constrained problems.

In the remaining part of this section we describe an adaptation to Problem (1.1) of the nonmonotone stabilization (NMS) technique introduced in [28], which has proven to be very effective in unconstrained optimization. We refer the interested reader to [28] for a more detailed discussion of this approach; here we just note that the NMS technique includes different strategies for enforcing global convergence without requiring a monotone reduction of the objective function. This may be reasonable in many situations. For example, if the sequence $\{\|d^k\|\}$ goes to zero, then, by Hypothesis 2(d), the corresponding sequence of points $\{x^k\}$ converges to a stationary point. A decrease of the norm of d^k is then a good measure for checking the convergence of the sequence $\{x^k\}$ to a stationary point.

The basic steps of the NMS technique can be roughly summarized in the following way. As long as the norm of the search direction is sufficiently reduced, set $x^{k+1} = x^k + d^k$ without evaluating the objective function. However, to ensure convergence to stationary points, a line search is performed if the norm of the search direction doesn't decrease enough. Furthermore, the objective function is evaluated at least every $\mathcal{N}$ iterations, and a backtrack occurs if its value is greater than a reference value $\mathcal{R}$ ($\geq f(x^k)$).

The steps of the NMS technique are presented below in detail.

NMS ALGORITHM.

Data: Choose $x^0 \in K$, $\Delta_0 \geq 0$, $\beta \in (0, 1)$, $\sigma \in (0, 1/2)$, $\mathcal{N} \in \mathbb{N}$ and $M \in \mathbb{N}_0$.

Initialization: Set $k = 0$, $j = 0$, and $\Delta = \Delta_0$.
Compute $f(x^0)$, and set $\mathcal{R}^j = f(x^0)$, $\ell(j) = k$, $m(j) = 0$ and $y^j = x^0$.

Iteration-k:

- Step 1:* Compute d^k satisfying Hypothesis 2. If $\|d^k\| = 0$ STOP.
- Step 2:* If $k = \ell(j) + \mathcal{N}$, compute $f(x^k)$, and
 - (a) If $f(x^k) \geq \mathcal{R}^j$, replace x^k by y^j , set $k = \ell(j)$, and go to Step 4;
 - (b) If $f(x^k) < \mathcal{R}^j$, set $j = j + 1$, $y^j = x^k$, $\ell(j) = k$, compute $\mathcal{R}^j$, and if $\|d^k\| \leq \Delta$, set $\alpha^k = 1$, $\Delta = \beta \Delta$, and go to Step 5; otherwise, go to Step 4;
- Step 3:* If $\|d^k\| \leq \Delta$, set $\alpha^k = 1$, $\Delta = \beta \Delta$, and go to Step 5;
If $\|d^k\| > \Delta$, compute $f(x^k)$ if necessary, and
 - (a) If $f(x^k) \geq \mathcal{R}^j$, replace x^k by y^j , set $k = \ell(j)$, and go to Step 4;
 - (b) If $f(x^k) < \mathcal{R}^j$, set $j = j + 1$, $y^j = x^k$, $\ell(j) = k$, compute $\mathcal{R}^j$, and continue;
- Step 4:* Find the smallest integer $i = 0, 1, \dots$ such that $f(x^k + 2^{-i} d^k) \leq \mathcal{R}^j + \sigma 2^{-i} \nabla f(x^k)^T d^k$.

Set $\alpha^k = 2^{-i}$, $j = j + 1$, $y^j = x^k + \alpha^k d^k$, $\ell(j) = k + 1$, and compute $\mathcal{R}^j$;
 Step 5: Set $x^{k+1} = x^k + \alpha^k d^k$, $k := k + 1$.

In order to complete the description of the algorithm we need only to explain how the reference value $\mathcal{R}^j$ is computed. To this end, we note that the index j is incremented whenever we have to evaluate the function value; in such an event the algorithm is forced to find a point in which the function value is smaller than the current reference value $\mathcal{R}^j$. This point is recorded as y^j and the iteration in which such an event occurred is recorded in $\ell(j)$, so that $y^j = x^{\ell(j)}$.

The reference value $\mathcal{R}^j$ is initially set equal to $f(x^0)$. Then, whenever necessary, it is updated by taking into account a number $m(j) \leq M$ of previous function values. The updating scheme takes the following form:

Compute $\mathcal{R}^j$:

Let $m(j)$ be a nonnegative integer such that

$$m(j) \leq \min(m(j-1) + 1, M).$$

Then choose the value $\mathcal{R}^j$ to satisfy

$$(2.3) \quad f(y^j) \leq \mathcal{R}^j \leq \max_{0 \leq i \leq m(j)} f(y^{j-i}).$$

The NMS algorithm is quite a general scheme and includes most of the line search techniques that have been used in unconstrained optimization. For example, if we set $M = 0$ and $\Delta_0 = 0$ we obtain the Armijo stabilization technique considered before.

In the same way as for the Armijo stabilization technique, it is not difficult to see that, by employing Hypotheses 1 and 2, we can adapt the proofs of the convergence of the NMS algorithm in the unconstrained case to establish the following result for the box constrained problem.

THEOREM 2.2. *If Hypotheses 1 and 2 hold, then the sequence $\{x^k\}$ generated by the NMS algorithm has at least a limit point. Furthermore,*

- (a) *every limit point of the sequence $\{x^k\}$ is a stationary point of Problem (1.1);*
- (b) *no limit point of the sequence $\{x^k\}$ is a local maximum of Problem (1.1).*

The proof of this theorem is almost a verbatim repetition of the proofs contained in [28] and we therefore omit it. The interested reader can find it in [38].

3. The search direction. In this section we show how to define at each feasible point x a search direction d which satisfies Hypothesis 2 and can be used in connection with the NMS algorithm described in the previous section.

The calculation of the search direction is divided into two parts. First, we estimate which bounds are active at the solution and define the corresponding search direction components. For example, if x^k is the current point and we estimate that at the optimal solution the i th component is at its lower bound l_i , we set $d_i^k = l_i - x_i^k$ so that in the next iteration we have $x_i^{k+1} = l_i$, provided that $\alpha^k = 1$. After doing this, we calculate the remaining part of the search direction. This is accomplished by solving a strictly convex quadratic subproblem in the space of the estimated free variables (corresponding to the bounds that have not been estimated to be active).

In order to introduce the procedure that estimates the active bounds, let $\bar{x} \in \mathbb{R}^n$ be a stationary point of Problem (1.1), and consider the associated active constraint

sets

$$(3.1) \quad \bar{L} := \{i: l_i = \bar{x}_i\}, \quad \bar{U} := \{i: \bar{x}_i = u_i\}.$$

Furthermore, let

$$\bar{F} := \{1, \dots, n\} \setminus (\bar{L} \cup \bar{U})$$

be the set of the free variables. By using this notation, the conditions (1.3) can be stated in the following form:

$$(3.2) \quad \begin{cases} \nabla f_i(\bar{x}) \geq 0 & \forall i \in \bar{L}, \\ \nabla f_i(\bar{x}) = 0 & \forall i \in \bar{F}, \\ \nabla f_i(\bar{x}) \leq 0 & \forall i \in \bar{U}. \end{cases}$$

Then, it seems fairly natural to define the following approximations $L(x)$, $F(x)$, and $U(x)$ to $\bar{L}$, $\bar{F}$, and $\bar{U}$, respectively,

$$(3.3) \quad \begin{aligned} L(x) &:= \{i : x_i \leq l_i + a_i(x) \nabla f_i(x)\}, \\ U(x) &:= \{i : x_i \geq u_i + b_i(x) \nabla f_i(x)\}, \\ F(x) &:= \{1, \dots, n\} \setminus L(x) \cup U(x), \end{aligned}$$

where $a_i(x)$ and $b_i(x)$, $i = 1, \dots, n$ are nonnegative continuous functions bounded from above on K , such that if $x_i = l_i$ or $x_i = u_i$ then $a_i(x) > 0$ or $b_i(x) > 0$, respectively. The following result shows that $L(x)$, $F(x)$, and $U(x)$ are indeed good estimations of $\bar{L}$, $\bar{F}$, and $\bar{U}$, respectively.

THEOREM 3.1. *For any feasible x , $L(x) \cap U(x) = \emptyset$. Furthermore, if $\bar{x}$ is a stationary point of Problem (1.1) where strict complementarity holds, then there exists a neighborhood of $\bar{x}$ such that for every feasible point x in this neighborhood we have*

$$L(x) = \bar{L}, \quad F(x) = \bar{F}, \quad U(x) = \bar{U}.$$

Proof. The first assertion follows by observing that if x is feasible and $x \in L(x)$ then $\nabla f_i(x) \geq 0$, whence, by the definition of $U(x)$, $x \notin U(x)$. Obviously we can exchange the role of $L(x)$ and $U(x)$ so that $L(x) \cap U(x) = \emptyset$. The second part easily follows by taking into account the optimality conditions (3.2), the nonnegativity of the functions $a_i(x)$ and $b_i(x)$, and by using continuity arguments. $\square$

The results of this theorem are fairly strong and yet it should be remarked that they can be established in a very simple way. Another noteworthy point is that the estimates (3.3) are completely decoupled from the algorithm that is used to solve Problem (1.1), thus allowing much freedom in the definition of the overall algorithm.

The estimates (3.3) can be related to some previous proposals in the literature. They are reminiscent of the technique first proposed in [18] for the case of nonlinear constraints and further employed in [19, 25]. We exploit the particularly simple structure of the feasible set K and the fact that the points produced by the algorithm are feasible to simplify the original proposal and obtain a very simple and yet effective tool. The estimates (3.3) can also be seen as a modification and a generalization of the guessing technique proposed in [3]. This generalization allows more flexibility and appears more naturally linked to the problem data. Furthermore, we are able to prove stronger results than those achieved in [3]. On the contrary, the properties established

in Theorem 3.1 are similar to those proved for the fixing technique introduced in [12] (see also [30]) and in [7]. However, these techniques are more complex than the one considered here.

Now let $x^k \in K$ be the current point at iteration k . Consider the sets $L^k = L(x^k)$, $U^k = U(x^k)$, and $F^k = F(x^k)$ and define the direction $d^k = (d_{L^k}^k, d_{F^k}^k, d_{U^k}^k)$ by

$$(3.4) \quad d_i^k = l_i - x_i^k, \quad i \in L^k;$$

$$(3.5) \quad d_i^k = u_i - x_i^k, \quad i \in U^k;$$

and set $d_{F^k}^k$ to be the optimal solution of the following strictly convex quadratic program

$$(3.6) \quad \begin{aligned} \min m(d_{F^k}) &= \nabla f_{F^k}(x^k)^T d_{F^k} + \frac{1}{2} d_{F^k}^T B_{F^k}^k d_{F^k} \\ \text{s. t.} \quad &l_{F^k} - x_{F^k}^k \leq d_{F^k} \leq u_{F^k} - x_{F^k}^k, \end{aligned}$$

where $B_{F^k}^k$ is a symmetric positive definite matrix.

In the remaining part of this section we show that if the standard Assumption 1 is fulfilled, then the direction $d^k, k = 1, 2, \dots$, defined by (3.4)–(3.6) satisfies Hypothesis 2 stated in the previous section.

ASSUMPTION 1. *There exists positive scalars ρ_1, ρ_2 such that any matrix $B_{F^k}^k, k = 1, 2, \dots$, satisfies*

$$\rho_1 \|z\|^2 \leq z^T B_{F^k}^k z \leq \rho_2 \|z\|^2 \text{ for all nonzero } z \in \mathbb{R}^{|F^k|}.$$

We first observe that Hypothesis 2(a) is easily seen to be satisfied by construction. The following result shows that Hypothesis 2(b) also holds.

THEOREM 3.2. *If Assumption 1 holds, $x^k \in K$ and d^k is the direction defined by (3.4)–(3.6), then Hypothesis 2(b) is satisfied with $p = 2$.*

Proof. Since $B_{F^k}^k$ is a symmetric positive definite matrix, then $d_{F^k}^k$ is the optimal solution of (3.6) if and only if

$$\begin{aligned} l_i - x_i^k = (d_{F^k}^k)_i &\implies (\nabla f_{F^k}(x^k) + B_{F^k}^k d_{F^k}^k)_i \geq 0, \\ l_i - x_i^k < (d_{F^k}^k)_i < u_i - x_i^k &\implies (\nabla f_{F^k}(x^k) + B_{F^k}^k d_{F^k}^k)_i = 0, \\ (d_{F^k}^k)_i = u_i - x_i^k &\implies (\nabla f_{F^k}(x^k) + B_{F^k}^k d_{F^k}^k)_i \leq 0. \end{aligned}$$

Since $x^k \in K$, each lower (upper) bound of the quadratic program (3.6) is negative (positive), whence

$$\begin{aligned} \nabla f_{F^k}(x^k)^T d_{F^k}^k + \frac{1}{2} d_{F^k}^{k^T} B_{F^k}^k d_{F^k}^k &\leq \nabla f_{F^k}(x^k)^T d_{F^k}^k + d_{F^k}^{k^T} B_{F^k}^k d_{F^k}^k \\ &= (\nabla f_{F^k}(x^k) + B_{F^k}^k d_{F^k}^k)^T d_{F^k}^k \leq 0. \end{aligned}$$

Then, by Assumption 1, we have that

$$(3.7) \quad \nabla f_{F^k}(x^k)^T d_{F^k}^k \leq -\rho_1 \|d_{F^k}^k\|^2.$$

Now we prove that there exists a positive scalar γ_i such that

$$(3.8) \quad \nabla f_i(x^k) d_i^k \leq -\gamma_i (d_i^k)^2,$$

for each $i \in L^k \cup U^k$. If $d_i^k = 0$ the inequality holds trivially. So suppose that $d_i^k \neq 0$. We only show the inequality for $i \in L^k$, since the case of $i \in U^k$ is analogous.

Since $x^k \in K$ and $d_i^k = l_i - x_i^k$ for each $i \in L^k$, then all nonzero d_i^k must be negative. Then it follows from the definition of the set L^k that

$$(3.9) \quad a_i(x^k) \nabla f_i(x^k) \geq -d_i^k.$$

Now $a_i(x^k) > 0$, because if $a_i(x^k) = 0$ (and $i \in L^k$) then $x_i^k = l_i$ and this implies $d_i^k = 0$. So $a_i(x^k) > 0$, $d_i^k < 0$ and hence, from (3.9), we get

$$\nabla f_i(x^k) d_i^k \leq -\frac{1}{a_i(x^k)} (d_i^k)^2.$$

But $a_i(x^k)$ is bounded from above on K , whence there exists $s_i \geq \sup_{l \leq x \leq u} a_i(x) > 0$, and (3.8) holds with $\gamma_i = 1/s_i$. The theorem now easily follows from (3.7) and (3.8). $\square$

Next, we prove that Hypothesis 2(c) is also satisfied.

THEOREM 3.3. *If Assumption 1 holds, $x^k \in K$, and d^k is the direction defined by (3.4)–(3.6), then*

$$d^k = 0 \iff x^k \text{ is a stationary point of } f \text{ on } K.$$

Proof. Let $d^k = 0$. If $i \in L^k$ then, by (3.3) and (3.4), we have

$$0 = d_i^k = l_i - x_i^k \geq -a_i(x^k) \nabla f(x^k)_i.$$

But, since $x_i^k = l_i$, $a_i(x^k) > 0$, whence $\nabla f_i(x^k) \geq 0$. On the other hand, for each $i \in U^k$ we have

$$0 = d_i^k = u_i - x_i^k \leq -b_i(x^k) \nabla f_i(x^k),$$

whence $\nabla f_i(x^k) \leq 0$.

Therefore, to establish that x^k is a stationary point of f on K it is sufficient to prove that $l_i < x_i^k < u_i$ and $\nabla f_i(x^k) = 0$, for each $i \in F^k$. If $i \in F^k$, we have

$$(3.10) \quad x_i^k > l_i + a_i(x^k) \nabla f_i(x^k), \quad x_i^k < u_i + b_i(x^k) \nabla f_i(x^k).$$

Suppose that there exists an $i \in F^k$ such that $\nabla f_i(x^k) < 0$. Then for sufficiently small $\epsilon > 0$, the vector $\tilde{d}_{F^k}$ defined by

$$\tilde{d}_j = \begin{cases} 0 & \text{if } j \in F^k \setminus \{i\}, \\ \epsilon & \text{if } j = i \end{cases}$$

satisfies $l_{F^k} - x_{F^k}^k \leq \tilde{d}_{F^k} \leq u_{F^k} - x_{F^k}^k$, i.e., it is feasible to (3.6), and

$$m(\tilde{d}_{F^k}) = \nabla f_i(x^k) \epsilon + \frac{1}{2} \epsilon^2 B_{ii}^k < 0,$$

which contradicts the assumption that the direction defined by (3.4)–(3.6) is $d^k = 0$. This is impossible since $d_{F^k}^k = 0$ is the optimal solution of (3.6). We could prove in a similar way that $\nabla f_i(x^k)$ cannot be positive. Hence, $\nabla f_i(x^k) = 0$ for each $i \in F^k$. Furthermore, $l_i < x_i^k < u_i$ by (3.10) and this proves that x^k is a stationary point of f on K .

Now suppose that x^k is a stationary point of f on K . Hence, it follows from (1.3) and (3.2) that

$$L^k = \{i: x_i^k = l_i\}, \quad F^k = \{i: l_i < x_i^k < u_i\}, \quad U^k = \{i: x_i^k = u_i\}.$$

Therefore, $d_{L^k} = d_{U^k} = 0$ by (3.4)–(3.5). On the other hand, $d = 0$ is a feasible solution of the quadratic program (3.6). Since $\nabla f_{F^k}(x^k) = 0$ and $B_{F^k}^k$ is a positive definite matrix,

$$m(d_{F^k}) = \frac{1}{2} d_{F^k}^T B_{F^k}^k d_{F^k} \geq 0.$$

Hence, $d_{F^k} = 0$ is the optimal solution of the quadratic program (3.6) and $d^k = 0$. $\square$

The next theorem proves that Hypothesis 2(d) is satisfied.

THEOREM 3.4. *Suppose that Assumption 1 is satisfied. Assume that $x^k \rightarrow \bar{x}$ and that $d^k \rightarrow 0$, where d^k is the direction defined by (3.4)–(3.6). Then $\bar{x}$ is a stationary point of Problem (1.1).*

Proof. Since the number of distinct sets L^k, U^k , and F^k is finite and Assumption 1 holds, we can assume, without loss of generality,

$$L^k = L, \quad F^k = F, \quad U^k = U, \quad B_{F^k}^k \rightarrow \bar{B}_F.$$

Since $x^k \in K$, we obviously have $l \leq \bar{x} \leq u$. Furthermore, $d_L^k \rightarrow 0$ and $d_U^k \rightarrow 0$ imply, taking into account the definition of L and U , that

$$\bar{x}_L = l_L, \nabla f_L(\bar{x}) \geq 0 \quad \text{and} \quad \bar{x}_U = u_U, \nabla f_U(\bar{x}) \leq 0.$$

So it remains to prove that $\nabla f_F(\bar{x}) = 0$. We recall that d_F^k is the solution of the quadratic program

$$\begin{aligned} & \min \nabla f_F(x^k)^T d_F + \frac{1}{2} d_F^T B_F^k d_F \\ & \text{s. t. } l_F - x_F^k \leq d_F \leq u_F - x_F^k. \end{aligned}$$

Since $d^k \rightarrow 0$, the continuity of the optimal solution of a strictly convex quadratic program under perturbations (see Proposition 6.1 of [17]) implies that zero is the optimal solution of

$$\begin{aligned} & \min \nabla f_F(\bar{x})^T d_F + \frac{1}{2} d_F^T \bar{B}_F d_F \\ & \text{s. t. } l_F - \bar{x}_F \leq d_F \leq u_F - \bar{x}_F. \end{aligned}$$

Hence, $\nabla f_F(\bar{x}) = 0$ by following reasonings similar to those used in the proof of Theorem 3.3. $\square$

To conclude this section we note that we can employ the search direction d^k computed by (3.4)–(3.5) and (3.6) in connection with the NMS technique described in the previous section to obtain convergence towards a stationary point of Problem (1.1). The resulting algorithm can be summarized as follows.

ACTIVE SET NONMONOTONE (ASNM) ALGORITHM.

Data: Choose $x^0 \in K$.

Initialization: Set $k = 0$.

Iteration-k: Find L^k, U^k , and F^k by (3.3);
 Find $d^k = (d_{L^k}^k, d_{F^k}^k, d_{U^k}^k)$ by (3.4)–(3.5) and (3.6);
 If $d^k = 0$, STOP: x^k is a stationary point of the Problem (1.1);
 Set $x^{k+1} = x^k + \alpha^k d^k$ and $k := k + 1$,
 where α^k is computed by the NMS technique.

4. Local analysis. In this section we study the local behavior of the active set Newton's algorithm defined in the previous section. We assume that at each iteration $B_{F^k}^k = \nabla^2 f_{F^k}(x^k)$, and we also assume that the following standard condition is satisfied.

ASSUMPTION 2.

- (a) $\{x^k\} \rightarrow \bar{x}$.
- (b) *Strict complementarity holds at $\bar{x}$.*
- (c) *The KKT second-order sufficient conditions for optimality hold at $\bar{x}$, i.e.,*

$$z^T \nabla^2 f_{\bar{F}}(\bar{x}) z > 0 \text{ for all nonzero } z \in \mathbb{R}^{|\bar{F}|}.$$

- (d) *The second-order derivatives of f are Lipschitz continuous.*

We now show that under Assumption 2 the convergence rate of $\{x^k\}$ to $\bar{x}$ is at least superlinear. To do this, we first prove that there exists $\bar{k}$ such that $\alpha^k = 1$ for all $k \geq \bar{k}$. Then we exploit this result to establish the convergence rate.

LEMMA 4.1. *If Assumption 2 holds then there exists $\bar{k}$ such that $\alpha^k = 1$ for all $k \geq \bar{k}$.*

Proof. By Theorem 3.1 we can assume, without loss of generality, that $L(x^k) = \bar{L}$, $F(x^k) = \bar{F}$, and $U(x^k) = \bar{U}$. For brevity, in the sequel we indicate the set $\bar{L} \cup \bar{U}$ by $\bar{A}$. We consider two distinct situations, namely, $\|x_{\bar{A}}^k - \bar{x}_{\bar{A}}\| = o(\|x^k - \bar{x}\|)$ and $\|x_{\bar{A}}^k - \bar{x}_{\bar{A}}\| \neq o(\|x^k - \bar{x}\|)$.¹ In either case we will prove that the unit stepsize is eventually attained.

First, let us focus on the case $\|x_{\bar{A}}^k - \bar{x}_{\bar{A}}\| = o(\|x^k - \bar{x}\|)$ and consider the following quadratic programming problem:

$$(4.1) \quad \min \tilde{m}(d_{F^k}) = \nabla f_{\bar{F}}(\bar{x}_{\bar{L}}, x_{\bar{F}}^k, \bar{x}_{\bar{U}})^T d_{\bar{F}} + \frac{1}{2} d_{\bar{F}}^T \nabla^2 f_{\bar{F}}(\bar{x}_{\bar{L}}, x_{\bar{F}}^k, \bar{x}_{\bar{U}}) d_{\bar{F}} \\ \text{s. t.} \quad l_{\bar{F}} - x_{\bar{F}}^k \leq d_{\bar{F}} \leq u_{\bar{F}} - x_{\bar{F}}^k.$$

For k sufficiently large, the quadratic problem (4.1) is strictly convex by Assumption 2(a) and 2(c). Then, by classical results on the stability of strictly convex quadratic programs (see, e.g., [16, Theorem 2.1]) and by the Lipschitz continuity of second-order derivatives of f , there is a scalar $c_1 \geq 0$ such that

$$(4.2) \quad \|d_{\bar{F}}^k - \tilde{d}_{\bar{F}}^k\| \leq c_1 \|(x_{\bar{L}}^k, x_{\bar{F}}^k, x_{\bar{U}}^k) - (\bar{x}_{\bar{L}}, x_{\bar{F}}^k, \bar{x}_{\bar{U}})\| = c_1 \|x_{\bar{A}}^k - \bar{x}_{\bar{A}}\|,$$

¹If $\{a^k\}$ and $\{b^k\}$ are two sequences of real numbers converging to 0, we write

$$a^k = o(b^k) \quad \text{if} \quad \limsup_{k \rightarrow \infty} \frac{a^k}{b^k} = 0$$

and

$$a^k = O(b^k) \quad \text{if} \quad 0 < \liminf_{k \rightarrow \infty} \left| \frac{a^k}{b^k} \right| \quad \text{and} \quad \limsup_{k \rightarrow \infty} \left| \frac{a^k}{b^k} \right| < \infty.$$

where $\tilde{d}_F^k$ is the unique solution of (4.1) and d_F^k is the solution of the quadratic problem (3.6) involved in the k th step of the algorithm.

Furthermore, by the strict complementarity assumption, none of the bounds in (4.1) is eventually active at the solution, so that

$$\tilde{d}_F^k = -(\nabla^2 f_{\bar{F}}(\bar{x}_{\bar{L}}, x_{\bar{F}}^k, \bar{x}_{\bar{U}}))^{-1} \nabla f_{\bar{F}}(\bar{x}_{\bar{L}}, x_{\bar{F}}^k, \bar{x}_{\bar{U}})$$

and $\tilde{d}_F^k$ is just the Newton direction for solving the (nonsingular) system of equations (in the $x_{\bar{F}}$ variable) $\nabla f_{\bar{F}}(\bar{x}_{\bar{L}}, x_{\bar{F}}, \bar{x}_{\bar{U}}) = 0$. Hence, from classical results on Newton's method we have that

$$(4.3) \quad \lim_{k \rightarrow \infty} \frac{\|x_F^k + \tilde{d}_F^k - \bar{x}_{\bar{F}}\|}{\|x^k - \bar{x}\|} = 0.$$

From (4.2), (4.3), and $\|x_A^k - \bar{x}_A\| = o(\|x^k - \bar{x}\|)$ we can now conclude that

$$(4.4) \quad \begin{aligned} \lim_{k \rightarrow \infty} \frac{\|x^k + d^k - \bar{x}\|}{\|x^k - \bar{x}\|} &= \lim_{k \rightarrow \infty} \frac{\|x_F^k + d_F^k - \bar{x}_{\bar{F}}\|}{\|x^k - \bar{x}\|} \\ &\leq \lim_{k \rightarrow \infty} \frac{\|x_F^k + \tilde{d}_F^k - \bar{x}_{\bar{F}}\|}{\|x^k - \bar{x}\|} + \lim_{k \rightarrow \infty} \frac{\|d_F^k - \tilde{d}_F^k\|}{\|x^k - \bar{x}\|} = 0. \end{aligned}$$

As it is shown, e.g., in [23], (4.4) implies

$$(4.5) \quad \lim_{k \rightarrow \infty} \frac{\|d^k\|}{\|x^k - \bar{x}\|} = 1,$$

$$(4.6) \quad \|x^k + d^k - \bar{x}\| = o(\|d^k\|),$$

which will be used later in the proof. Furthermore, by using Taylor's expansion, we can now write

$$(4.7) \quad f(x^k + d^k) = f(\bar{x}) + \nabla f(\bar{x})^T (x^k + d^k - \bar{x}) + \frac{1}{2} (x^k + d^k - \bar{x})^T H^k (x^k + d^k - \bar{x})$$

$$(4.8) \quad f(x^k) = f(\bar{x}) + \nabla f(\bar{x})^T (x^k - \bar{x}) + \frac{1}{2} (x^k - \bar{x})^T R^k (x^k - \bar{x}),$$

where $H^k = \nabla^2 f(z^k)$ for some $z^k \in (\bar{x}, x^k + d^k)$ and $R^k = \nabla^2 f(w^k)$ for some $w^k \in (\bar{x}, x^k)$. Subtracting (4.8) from (4.7) and taking into account (4.6), we obtain

$$f(x^k + d^k) - f(x^k) = \nabla f(\bar{x})^T d^k - \frac{1}{2} (x^k - \bar{x})^T R^k (x^k - \bar{x}) + o(\|d^k\|^2).$$

But recalling the optimality conditions (1.3) and the strict complementarity assumption, we have that

$$(4.9) \quad \nabla f(\bar{x})^T d^k = (\nabla f_{\bar{L}}(\bar{x})^T, 0_{\bar{F}}^T, \nabla f_{\bar{U}}(\bar{x})^T) \begin{pmatrix} \bar{x}_{\bar{L}} - x_{\bar{L}}^k \\ d_F^k \\ \bar{x}_{\bar{U}} - x_{\bar{U}}^k \end{pmatrix} < 0$$

so that

$$f(x^k + d^k) - f(x^k) - \frac{1}{2} \nabla f(x^k)^T d^k \leq -\frac{1}{2} (x^k - \bar{x})^T R^k (x^k - \bar{x}) - \frac{1}{2} \nabla f(x^k)^T d^k + o(\|d^k\|^2).$$

The right-hand side of the previous inequality is equal to

$$-\frac{1}{2}(x^k + d^k - \bar{x})^T R^k(x^k - \bar{x}) + \frac{1}{2}d^{kT} R^k(x^k - \bar{x}) - \frac{1}{2}\nabla f(x^k)^T d^k + o(\|d^k\|^2).$$

By using (4.5) and (4.6) we then get

$$\begin{aligned} f(x^k + d^k) - f(x^k) - \frac{1}{2}\nabla f(x^k)^T d^k &\leq -\frac{1}{2}d^{kT} [\nabla f(x^k) - R^k(x^k - \bar{x})] + o(\|d^k\|^2) \\ &= o(\|d^k\|^2), \end{aligned}$$

where the last equality follows from [23, Proposition 2.4] and (4.6). The assertion on the acceptance of the stepsize of one now follows from the fact that $\gamma < 1/2$ and Theorem 3.2.

We now examine the case $\|x_{\bar{A}}^k - \bar{x}_{\bar{A}}\| \neq o(\|x^k - \bar{x}\|)$. We assume, without loss of generality, that

$$(4.10) \quad \|x_{\bar{A}}^k - \bar{x}_{\bar{A}}\| \geq c_2 \|x^k - \bar{x}\|$$

for some positive c_2 . In fact, if, for some subsequence K , $\|x_{\bar{A}}^k - \bar{x}_{\bar{A}}\| = o(\|x^k - \bar{x}\|)$, $k \in K$, then, as in the previous case we can show that eventually $\alpha^k = 1$ for $k \in K$ and sufficiently large and we could analyze the behavior of the algorithm only on the remaining subsequence of indices. So, we assume that (4.10) holds.

We first recall that (4.9) still holds. Furthermore, again by the optimality conditions (1.3) and the strict complementarity assumption, and taking into account (4.10), we also have

$$(4.11) \quad \left| (\nabla f_{\bar{L}}(\bar{x})^T, 0_{\bar{F}}^T, \nabla f_{\bar{U}}(\bar{x})^T) \begin{pmatrix} \bar{x}_{\bar{L}} - x_{\bar{L}}^k \\ d_{\bar{F}}^k \\ \bar{x}_{\bar{U}} - x_{\bar{U}}^k \end{pmatrix} \right| \geq C \|x_{\bar{A}}^k - \bar{x}_{\bar{A}}\| = O(\|x^k - \bar{x}\|),$$

where the positive constant C is the smallest modulus of the components of the vector $\nabla f_{\bar{A}}(\bar{x})$. On the other hand, by the strict complementarity assumption, none of the bounds in (3.6) is active for k sufficiently large, so that

$$d_{\bar{F}}^k = -(\nabla^2 f_{\bar{F}}(x^k))^{-1} \nabla f_{\bar{F}}(x^k).$$

Since $\nabla f_{\bar{F}}(\bar{x}) = 0$ and the gradient of f is Lipschitz continuous, we then have that

$$(4.12) \quad \|d^k\| = O(\|x^k - \bar{x}\|).$$

As the Taylor's expansions (4.8) and (4.7) are still valid, subtracting (4.8) from (4.7) and adding to both sides $-\gamma \nabla f(x^k)^T d^k$, we obtain

$$\begin{aligned} f(x^k + d^k) &- f(x^k) - \gamma \nabla f(x^k)^T d^k = \frac{1}{2} \nabla f(\bar{x})^T d^k + \left(\frac{1}{2} - \gamma \right) \nabla f(\bar{x})^T d^k \\ (4.13) \quad &+ \gamma (\nabla f(\bar{x}) - \nabla f(x^k))^T d^k + \frac{1}{2} (x^k + d^k - \bar{x})^T H^k (x^k + d^k - \bar{x}) \\ &- \frac{1}{2} (x^k - \bar{x})^T R^k (x^k - \bar{x}). \end{aligned}$$

Now, by (4.12) it readily follows that the last three terms in the right-hand side of (4.13) are $o(\|x^k - \bar{x}\|)$. Hence, (4.13), (4.9), (4.11), (4.12), Theorem 3.2, and $\gamma < 1/2$ give

$$f(x^k + d^k) - f(x^k) - \gamma \nabla f(x^k)^T d^k \leq 0$$

for k large enough. This implies that eventually the stepsize of one is accepted, which concludes the proof. $\square$

THEOREM 4.2. *If Assumption 2 is satisfied, then the convergence rate of $\{x^k\}$ to $\bar{x}$ is quadratic.*

Proof. We first note that from Theorem 3.1 the sets L^k and U^k remain constant in the last iterations of the algorithm. Hence, $L(x^k) = \bar{L}$ and $U(x^k) = \bar{U}$ for all k sufficiently large. By Lemma 4.1 we have that, eventually,

$$(4.14) \quad d_L^k = 0, \quad x_L^k = l_{\bar{L}}, \quad d_U^k = 0, \quad x_U^k = u_{\bar{U}}.$$

Furthermore, by the complementarity slackness assumption, none of the bounds of the quadratic subproblem is eventually active. Hence, the algorithm reduces to Newton's method in the subspace of free variables. Since $\nabla^2 f_{\bar{F}}(\bar{x}_{\bar{L}}, \bar{x}_{\bar{F}}, \bar{x}_{\bar{U}})$ is positive definite by Assumption 2, the theorem follows from classical results on Newton's method. $\square$

We believe that superlinear convergence of the algorithm on degenerate problems cannot be proved. In fact, the curvature information on the subspace of the active degenerate variables may be lost if these variables are (correctly) estimated to be at their bounds at the solution, and this may prevent superlinear convergence. We think that it should be possible to modify the algorithm in order to guarantee superlinear convergence on degenerate problems; however, we leave this issue as a subject for future research.

5. Large-scale implementation. Each iteration of the algorithm requires the solution of a strictly convex quadratic program with box constraints of the following form:

$$(5.1) \quad \begin{aligned} \min m(d_F) &= \nabla f_F(x)^T d_F + \frac{1}{2} d_F^T B_F d_F \\ \text{s. t. } l_F - x_F &\leq d_F \leq u_F - x_F, \end{aligned}$$

where we drop the superscript k for sake of simplicity. It follows from the results of sections 3 and 4 that a good choice to achieve global and superlinear convergence is to set $B_F = \nabla^2 f_F(x) + D$, where D is a diagonal matrix with nonnegative diagonal elements such that $D = 0$ if and only if the smallest eigenvalue of $\nabla^2 f_F(x)$ is greater than a small positive quantity. Actually, if we exclude the computation of the function and gradient values, which have an unpredictable cost, the solution of the subproblem (5.1) and the calculation of D account for most of the computational work in the algorithm. Hence, special care must be used when dealing with these points in a large-scale setting.

In this section we discuss the linear algebra issues relevant to a large-scale implementation of the block principal pivoting algorithm for the solution of the quadratic subproblem (5.1) and how to choose the matrix D . Furthermore, we also consider the data structures employed in the implementation of the algorithm.

The block principal pivoting algorithm [29] is primarily a method for linear complementarity problems which can easily be modified to handle the presence of bounds. Its specialization to strictly convex box constrained quadratic programs is finitely convergent and usually effective in the large-scale case, see [29] where the reader may find a more detailed discussion of the method. Here we simply note that, at each iteration, a guess is made of the variables which are at their bounds or are free. Then, a reduced linear system is solved (the dimension of the linear system being equal to the number

of estimated free variables) and a candidate solution is calculated. If an optimality test is satisfied the procedure is terminated. Otherwise, the procedure is repeated. In this scheme, the most intensive task is the solution, at each inner iteration, of the (reduced) linear system. We note that the matrix associated to the various linear systems is always a principal submatrix of $B_F = \nabla^2 f_F(x) + D$.

5.1. Calculation of D and solution of linear systems. In this subsection we discuss how to build the diagonal matrix D and how to reduce the cost of the solution of the linear systems required by the block pivoting algorithm.

The most common way for finding a suitable matrix D consists of performing a *modified Cholesky factorization* of $\nabla^2 f_F(x)$. Two main algorithms have been proposed in the literature, [26] and [36]. Computational experience has shown that the method of Gill and Murray [26] is more reliable in our framework, when sparsity is exploited during the modified factorization of the Hessian matrix (see [37]). Therefore this is the procedure we have used to compute the matrix B_F . To understand how this modified factorization and the other operations required by the block pivoting algorithm can be performed efficiently, we have to go into some technical details about the technique adopted for solving linear systems.

In our work we basically use Harwell subroutine MA27 [21] to solve linear systems. This routine performs a sparse Cholesky factorization of a positive definite matrix to solve the associated linear system. It uses a direct multifrontal method and is well suited to large and sparse systems. The Cholesky factorization requires three phases: ordering, symbolic factorization, and numerical solution. In the ordering phase a pivoting order is found which tries to minimize the fill-in of the matrix factors to be computed in the subsequent phases. In the symbolic factorization phase the positions of the nonzeros in the Cholesky factors are determined and an adequate data structure is set up. Finally, the calculations are carried out in the numerical phase.

It is well known that the major burden in solving large linear systems is generally associated with the ordering phase. The ordering procedure only depends on the sparsity structure of the matrix (and not on the numerical values of its entries). Furthermore, it is important to note that once the ordering has been determined for a matrix, a good ordering for any of its principal submatrices can be easily derived.

We assume that the sparsity structure of the Hessian of the objective function is available before the initialization of the main algorithm. Then we proceed as follows.

1. We determine an ordering for the Hessian of the objective function before starting the main algorithm.

2. At the beginning of the solution of each box quadratic subproblem, we perform a (Gill–Murray) modified Cholesky factorization to obtain a diagonal matrix D such that $\nabla^2 f_F(x) + D$ is positive definite and we set $B_F = \nabla^2 f_F(x) + D$.

3. We use MA27 to solve each linear system required by the block pivoting algorithm.

4. In the first iteration of the block pivoting algorithm it is always assumed, as initial guess, that all the variables of the quadratic subproblem are free.

The following points should be noticed.

- The costly ordering phase is performed just once during the whole algorithm.
- A straightforward modification of MA27 is used to obtain the matrix D of point 2, while solving the first system of the block pivoting algorithm. The values of the parameters defined in [26] and employed in the modified Cholesky factorization are given by

$$\delta = 10^{-15} \max(1, \gamma) \quad \text{and} \quad \beta = \sqrt{\max(10^{-15}, \gamma)},$$

where γ is the largest absolute value of the diagonal elements of the projected Hessian $\nabla^2 f_{F^k}(x^k)$.

- The solution of each linear system required by the block pivoting algorithm (see point 3) and the calculation of the diagonal matrix (see point 2) only require the symbolic and numerical factorizations and solution phases of MA27.

Due to the choice indicated in point 4, the block pivoting algorithm always employs, as first candidate solution to the subproblem, the vector $-B_F^{-1}\nabla_F f(x)$. In turn, this implies that the quadratic subproblem is solved in just one (inner) iteration eventually. In fact, close enough to a stationary point where strict complementarity holds, Theorem 3.1 ensures that the set of free variables is correctly identified ($F^k = \bar{F}$). Since d^k converges to 0, the solution of the quadratic problem is then given by $-B_{\bar{F}}^{-1}\nabla_{\bar{F}} f(x)$. Furthermore, if the smallest eigenvalue of the matrix $\nabla^2 f_{\bar{F}}(\bar{x})$ is greater than a certain small positive quantity (which depends on δ and β), this implies that $D = 0$, see [26]. Note that different strategies for modifying the projected Hessian could be employed in order to ensure that $D = 0$ whenever the sequence $\{x^k\}$ is converging to a point where the second-order sufficient conditions are met (see [3]). This would guarantee theoretical quadratic convergence without any assumption on the smallest eigenvalue of $\nabla^2 f_{\bar{F}}(\bar{x})$. However, from a numerical point of view this could imply the solution of progressively more ill-conditioned problems so that the solution adopted here is generally preferred.

5.2. Other implementation issues. We have seen that a key element to reduce the computational cost of the algorithm is the determination of the sparsity structure of the Hessian of the objective function at the beginning of the algorithm. We store only the nonzero elements of the Hessian according to a column oriented scheme [20]. This requires three vectors **Hptc**, **Hrow**, and **Hval**, of dimension $n + 1$, nnz , and nnz , respectively, where nnz is the number of nonzero elements of the Hessian. The first two vectors characterize the sparsity structure of the Hessian, while the vector **Hval** contains the values of the nonzero elements at the current point. Hence, the computation of the sparsity structure of the Hessian of the objective function reduces to find the vectors **Hptc**, **Hrow**. In general, this structure is readily available for large-scale problems. Next, we describe a procedure to determine this structure when it is not known beforehand. Consider a (possibly random) feasible point and evaluate sequentially the columns of the Hessian at that point (analytically or by finite differences). Then take as zero all the elements which are smaller than a prespecified small number (typically the machine precision) and build the vectors **Hptc**, **Hrow** according to the information gained. This procedure can obviously lead to mistakes, with a deterioration of the overall efficiency, if the sparsity structure of the Hessian varies greatly from point to point. However, it typically gives very satisfactory results. This procedure is suitable for small-scale problems. The process can also be applied to large-scale problems, but it is clear that it is prohibitively expensive, unless the cost of the Hessian-vector product is very low. The automatic and cheap determination of the sparsity structure of the Hessian is the subject of current research (see, e.g., [9]) and we expect that reliable software will be available soon.

It is also important to note that the implementation of the algorithm does not require a subroutine to evaluate the Hessian. In fact, it only requires a subroutine that computes Hessian-vector products (analytically or by finite differences). Using this subroutine, it is always possible to reconstruct the full Hessian by performing n products with the vectors e_i of the canonical basis of $\mathbb{R}^n$. However, when dealing with large-scale problems, the Hessian is often sparse and well structured, so that few

selected products are generally needed for its computation. The problem of finding the minimum number of Hessian-vector products necessary to evaluate the Hessian can be cast as a graph coloring problem [11]. This is an NP hard problem, and several heuristics have been suggested for its solution [35], [11]. Here we employ the method proposed in [35] which the numerical experiments of [11] have shown to be quite efficient. We also remark that this procedure is carried out only once, before the algorithm starts.

6. A technique for generating test problems. In this section we describe a new technique for generating box constrained optimization problems with known characteristics. This technique allows us to control the number of active constraints and the degeneracy at a known local minimum point and will be used in the next section. Consider an unconstrained optimization problem

$$(6.1) \quad \min_{x \in \mathbb{R}^n} f(x),$$

where f is twice continuously differentiable. Let $\bar{x}$ be a known local minimum of this unconstrained problem. Let $\bar{F}$, $\bar{L}$, and $\bar{U}$ be the sets of indices of the variables that will be free, at a lower bound and at an upper bound at the solution of the box constrained optimization problem to be designed. Then we choose the vectors l and u to satisfy the following inequalities:

$$(6.2) \quad l_{\bar{L}} = \bar{x}_{\bar{L}} < u_{\bar{L}}, \quad l_{\bar{F}} < \bar{x}_{\bar{F}} < u_{\bar{F}}, \quad l_{\bar{U}} < \bar{x}_{\bar{U}} = u_{\bar{U}}.$$

In particular, we note that if some component of $u_{\bar{L}}$ or $u_{\bar{F}}$ is equal to $+\infty$ then the corresponding variable has no upper bound, while if some component of $l_{\bar{U}}$ or $l_{\bar{F}}$ is equal to $-\infty$ then the corresponding variable has no lower bound.

Now consider the following function:

$$(6.3) \quad g(x) = f(x) + \sum_{i \in \bar{L}} h_i(x_i) - \sum_{i \in \bar{U}} h_i(x_i),$$

where

$$(6.4) \quad h_i(x_i) = (x_i - \bar{x}_i)^{7/3} + \beta_i(x_i - \bar{x}_i) \quad \forall i \in \bar{L} \cup \bar{U}$$

for some nonnegative constants β_i , $i \in \bar{L} \cup \bar{U}$. Since the functions h_i , $i \in \bar{L} \cup \bar{U}$, are nondecreasing, then it follows from (6.2) and (6.3), that $\bar{x}$ is a local minimum of the following box constrained optimization problem

$$(6.5) \quad \min_{x \in \mathbb{R}^n: l \leq x \leq u} g(x).$$

Furthermore, $h'_i(\bar{x}_i) \geq 0$ for $i \in \bar{L} \cup \bar{U}$, so that it is easy to check that $\bar{x}$ is a stationary point of problem (6.5) if it is a stationary point of problem (6.1).

It is important to note that there is no restriction in the choice of the sets $\bar{F}$, $\bar{L}$ and $\bar{U}$ and on the values of $u_{\bar{L}}$, $u_{\bar{F}}$, $l_{\bar{U}}$, and $l_{\bar{F}}$. By changing these parameters we can control the number of bounds, their magnitude, and how many of them are active at $\bar{x}$.

Furthermore, it is easy to see that the Lagrange multipliers associated with the constraints $l_{\bar{L}} \leq x_{\bar{L}}$ and $x_{\bar{U}} \leq u_{\bar{U}}$ are β_i for $i \in \bar{L} \cup \bar{U}$, while the remaining multipliers are 0. Hence, we also have a complete control over the degeneracy of problem (6.5) at $\bar{x}$.

Because of the choice of h_i , the objective function g is at most twice continuously differentiable and the Hessian of g and of the underlying function f are equal at $\bar{x}$ so that second-order properties (e.g., ill-conditioning) of the original unconstrained problem are preserved as much as possible.

We note that in general we cannot guarantee that the problems generated do not have stationary points other than $\bar{x}$. However, if f is strictly convex, so is g , thus ensuring that $\bar{x}$ is the only solution of the problem generated.

7. Computational experience. In this section we examine the numerical behavior of the algorithm ASNM on a set of test problems. The algorithm has been coded in FORTRAN 77, on a SUN SPARC 10 (48 Mhz, 64 Mb RAM), using double precision storage and arithmetic (about 16 decimal places precision).

In our experiments we have chosen the following standard values for the parameters of the algorithm. In the NMS technique we have set $\beta = 0.9, \sigma = 0.5, M = 20$, and $\mathcal{N} = 20$. The parameter Δ_0 is computed automatically during the first line search, according to the procedure suggested in [28]. We note that this procedure may require some additional function evaluations. We have always taken $\mathcal{R}^j = \max_{0 \leq i \leq m(j)} f(y^{j-i})$, where $m(j) = \min[m(j-1) + 1, M]$ except when a restart occurs (Steps 2(a) and 3(a)) and when $F^k \neq F^{k-1}$, in which case we set $m(j) = 0$ (see [28] for some more details).

The stopping criterion is expressed in terms of the projected gradient $P(\nabla f(x^k))$, where, for $i = 1, \dots, n$,

$$P(\nabla f(x^k))_i = \begin{cases} \min(0, \nabla f_i(x^k)) & \text{if } x_i = l_i, \\ \nabla f_i(x^k) & \text{if } x_i \in (l_i, u_i), \\ \max(0, \nabla f_i(x^k)) & \text{if } x_i = u_i. \end{cases}$$

The algorithm terminates when $\|P(\nabla f(x^k))\|_\infty \leq 10^{-6}$. We have also included the additional stopping criterion $\|d^k\|_\infty \leq 10^{-10}$ which is justified by Theorems 3.3 and 3.4 and might be useful for highly degenerate problems. However, the algorithm has always stopped with the verification of the test on the projected gradient in all tests but one.

The bounds of the quadratic subproblem have been set to

$$(7.1) \quad \tilde{l}_i^k = \max(l_i - x_i^k, -1000), \quad \tilde{u}_i^k = \min(u_i - x_i^k, 1000) \quad \forall i \in F^k$$

whenever the projected Hessian $\nabla^2 f_{F^k}(x^k)$ is perturbed in the first stage of the block pivoting algorithm. Our experience has shown that this procedure prevents the algorithm from getting erroneous solutions due to ill-conditioning of the perturbed matrix. It can be easily seen that the convergence analysis remains valid with this safeguard.

Finally, before starting the algorithm, we employ the heuristic procedure proposed in [24] in order to determine a new starting point. The heuristic resets the starting point to the projection on the feasible set of

$$(7.2) \quad x^0 - \frac{2 \max(1, |f(x^0)|)}{\max(10^{-20}, \|\nabla f(x^0)\|^2)} \nabla f(x^0)$$

whenever the objective function value at this point is smaller than at x^0 . We found in [24] that this simple modification can, in some cases, reduce the number of iterations needed by iterative algorithms for solving bound constrained problems. This has also been confirmed by preliminary numerical tests on our algorithm, which we do not report here for sake of brevity.

The test problems used in our experiments are either described in the literature or generated according to the technique of section 6. In particular, we have used problems from the CUTE collection [5] and problems generated from unconstrained problems in the MINPACK-2 collection [1]. While the first collection is becoming widely used to assess the performance of general nonlinear optimization algorithms, the problems generated using the unconstrained MINPACK-2 test problems allow us to better and more systematically analyze the behavior of the algorithm and its characteristics.

7.1. CUTE test problems. Our first experiments have been performed on the set of all sparse nonquadratic box constrained problems from the CUTE collection that have second derivatives available and dimension between 500 and 2000. Some relevant data on the eight resulting problems are collected in Table 1, namely, the name (Name), the dimension (n), the number of lower and upper bounds, respectively, (lower/upper), and the number of Hessian by vector products necessary to evaluate the Hessian matrix (p). The problems EXPLIN, EXPLIN2, EXPQUAD, and QRTQUAD depend on a parameter whose value has been fixed at 100. In all the tests reported in this section the starting point x^0 is the one provided by CUTE to which, however, we have applied the heuristic (7.2).

TABLE 1
CUTE problems.

Name	n	lower/upper	p	Name	n	lower/upper	p
BDEXP	1000	1000/0	5	QRTQUAD	1200	100/100	3
EXPLIN	1200	1200/1200	3	MCCORMCK	1000	1000/1000	3
EXPLIN2	1200	1200/1200	3	NONSCOMP	1000	1000/1000	3
EXPQUAD	1200	100/100	3	LINVERSE	999	500/0	9

Table 2 presents the results obtained on these problems. In this and in the remaining tables $\#it$ is the number of iterations, $\#Bl$ is the total number of block pivoting inner iterations, and $\#f$ is the number of function evaluations. Additionally, we report the number of variables at the solution found by the algorithm which are at their lower ($\#L$) and upper ($\#U$) bounds, and the function value at the solution (f^*). Note that in our algorithm the number of gradient evaluations is always $\#f + 1$.

TABLE 2
CUTE problems: algorithm results with (I) $a_i(x) = b_i(x) = 10^{-6} \forall i$.

Name	$\#it$	$\#Bl$	$\#f$	$\#L$	$\#U$	f^*
BDEXP	15	15	14	0	0	$0.115D - 5$
EXPLIN	13	30	13	0	1150	$-.7192D8$
EXPLIN2	9	36	10	0	1181	$-.7200D8$
EXPQUAD	29	310	55	0	81	$-.3685D10$
QRTQUAD	23	303	48	0	50	$-.3685D10$
MCCORMCK	3	4	7	1	0	-913.7
NONSCOMP	7	7	7	486	0	$1D - 14$
LINVERSE	140	219	326	346	0	340.0

In the first set of experiments we have investigated the impact of different choices for functions a_i and b_i that are used in the definitions of the sets $L(x^k)$ and $U(x^k)$. It seems reasonable to try functions whose values decrease as the minimization proceeds, so that the closer to a stationary point the more accurate the identification is. Therefore, besides (I) $a_i(x) = b_i(x) = 10^{-6}$ which was used to obtain the results of Table 2,

we have also tried (II) $a_i(x) = b_i(x) = 10^{-3}$, (III) $a_i(x) = b_i(x) = 10^{-3}\|P(\nabla f(x))\|$, and (IV) $a_i(x) = b_i(x) = 10^{-6}\|P(\nabla f(x))\|$. The function values are safeguarded so that they never become too large or too small. The results are reported in Table 3 and show that options II, III, and IV do not in general lead to consistently better results.

TABLE 3
CUTE problems: different choices for a_i and b_i .

Name	(II)			(III)			(IV)		
	#it	#Bl	#f	#it	#Bl	#f	#it	#Bl	#f
BDEXP	15	15	14	15	15	14	15	15	14
EXPLIN	11	30	15	17	80	30	10	29	13
EXPLIN2	32	113	43	13	58	26	9	47	14
EXPQUAD	50	386	95	91	480	149	72	456	113
QRTQUAD	32	182	126	32	132	129	28	130	96
MCCORMCK	3	4	7	3	4	7	3	4	7
NONSCOMP	7	7	7	7	7	7	7	7	7
LINVERSE	139	218	322	216	305	419	137	214	326

We have also tried some more choices for the functions $a(x)$ and $b(x)$ but without any success. Hence, we set $a_i(x) = b_i(x) = 10^{-6}$, $i = 1, \dots, n$ in the remaining numerical experiments. However, we are convinced that the flexibility offered by the guessing technique could be better exploited. The results reported merely indicate that a correct choice for the functions a_i and b_i requires some care and cannot be obtained by elementary choices. Since we are only interested in showing the viability of our approach, we postpone the study of a better choice to a more extensive numerical study.

As stated before, the guessing technique included in the algorithm allows, at each iteration, to add or drop an arbitrary number of variables from the active set ($L^k \cup U^k$), and by Theorem 3.1 the active constraints with positive multipliers will eventually be correctly determined. In the next experiments we investigate the practical performance of this strategy. We display in the first six columns of Table 4 the first iteration in which a certain percentage of variables are correctly identified. In the last column we report the percentage of bounded variables whose status (free or at a bound) is correctly identified at the heuristic starting point.

TABLE 4
CUTE problems: accuracy of the guessing strategy.

Name	100%	99%	98%	95%	90%	80%	% at x^0
BDEXP	0	0	0	0	0	0	100
EXPLIN	8	7	6	1	1	1	4.2
EXPLIN2	5	4	1	1	1	1	1.6
EXPQUAD	24	24	24	23	23	18	19
QRTQUAD	20	19	19	18	18	18	52
MCCORMCK	1	0	0	0	0	0	99.9
NONSCOMP	7	7	7	7	7	7	51.4
LINVERSE	133	104	89	62	53	36	28.8

The results show that the guessing technique behaves well on this set of problems, and seems capable of identifying a large part of the correct active set at an early stage of the minimization process. The conclusion does not appear to hold for the NONSCOMP problem. However, this was expected since this problem is highly degenerate. The good quality of our guessing technique is also confirmed by the tests

reported in the next subsection. However, we remark that our conclusions are simply empirical; it is obviously possible to build *ad hoc* examples for which the guessing strategy proposed does not behave well. This is true for any guessing strategy which, as ours, is based just on local information, since the problem of identifying the correct active set is basically global. An example that caused difficulties to our algorithm can be found in the CUTE collection. Problem BIGGSB1 is a convex, quadratic problem with 999 variables with finite lower and upper bounds and one free variable. At the starting point all the variables but one are misclassified and only two variables change their status at each iteration; convergence to the solution was obtained after 499 iterations.

Finally we analyze execution times. In Table 5 we report the execution times of our implementation. In particular:

- T_a is the total time spent reading the Hessian sparsity structure and translating it into an adequate data structure, determining how many Hessian by vector products are needed to calculate the full Hessian and performing the ordering phase;
- T_r is the time required by the iterative process;
- T_{tot} is the total execution time, given by the sum of T_a and T_r ;
- T_f, T_g and T_H are the times spent evaluating f , ∇f and $\nabla^2 f$, respectively.

TABLE 5
CUTE problems: total and partial execution times.

Name	T_a	T_r	T_{tot}	T_f	T_g	T_H
BDEXP	.60	3.02	3.62	.20	.22	.76
EXPLIN	.35	.61	.96	.03	.04	.11
EXPLIN2	.34	.60	.94	.04	.02	.08
EXPQUAD	.85	63.23	64.08	.27	.25	.75
QRTQUAD	.85	95.82	96.67	.19	.21	.56
MCCORMCK	.41	.84	1.25	.06	.10	.17
NONSCOMP	.43	1.02	1.44	.07	.08	.20
LINVERSE	.97	65.32	66.29	3.52	4.05	27.27

As the time required for evaluating the Hessian is a small multiple of that associated with the computation of the function values, we come to the conclusion that the sparse techniques adopted to calculate the Hessian are viable, at least on these test problems. We also note that T_a can be a significant fraction of the entire computing time when few iterations are required, but it tends to become negligible when many iterations are needed. In this latter case the advantage of performing only once the costly ordering phase needed in the solution of the linear systems is apparent.

7.2. MINPACK-2 test problems. In this subsection we systematically study, on a small set of test problems, the algorithm sensitivity to various characteristics of the problems—namely, the number and distribution of the constraints, degeneracy, and the shape of the feasible region. The problems considered in this subsection are obtained by applying the generation technique of section 6 to some unconstrained minimization problems from the MINPACK-2 collection [1]. This test problem collection contains eight nonlinear minimization problems. We have excluded two convex quadratic problems, the Lennard–Jones clusters problem for which neither the sparsity structure nor the Hessian-vector product subroutine are provided, and the homogeneous superconductor problem, since there seems to be a flaw in the routine that gives the Hessian-vector product. Some information about the remaining problems

is collected in Table 6; we report the problem name and its original identification number in the MINPACK-2 collection, the dimension n , and the number p of the Hessian-vector products needed to calculate the Hessian matrix.

TABLE 6
MINPACK-2 problems.

Id	Problem	n	p
4.3	Minimal surfaces	1024	9
4.4	Optimal design with composite materials	1024	9
4.5	Inhomogeneous superconductor (1-D Ginzburg-Landau)	1000	3
4.7	Steady-state Combustion	1024	9

For each problem we first computed the unconstrained solution $\bar{x}$ required as input to the test problem generator. The results obtained are presented in Table 7. Additionally, in this table we present the infinity norm of the gradient at the last iterate ($\|\nabla f^*\|_\infty$). As expected in the unconstrained case, the number of block pivoting iterations is equal to the number of iterations of the ASNM algorithm, since the quadratic subproblems are unconstrained. The number of function evaluations, although low, is higher than what one could expect (see, for example, problems 4.5 and 4.7) because in the first iteration a particular line search is performed to determine a good value of Δ_0 , which requires, on average, from three to six extra function evaluations. This procedure is carried out only once and could be eliminated without great damage. However, we chose to use the NMS algorithm as a black box, without altering its features.

TABLE 7
MINPACK-2 problems: ASNM results.

Prob.	#it	#Bl	# f	f^*	$\ \nabla f^*\ $
4.3	5	5	11	1.42104	.5D-14
4.4	11	11	11	-0.01562	.4D-15
4.5	11	11	21	-8456.19	.3D-07
4.7	3	3	10	-5.60996	.6D-14

To describe in a short way the characteristics of the test problems generated we have used five parameters *bnd*, *lowbnd*, *actbnd*, *ndeg*, and *width*, whose meanings are described below.

bnd: This parameter can vary between 0 and 100 and determines the number of constraints of the problem. The maximum number of constraints is $2n$, and *bnd* gives the percentage of this maximum number that should be present in the problem. The way in which they are distributed among lower and upper bounds and how many of them are active at $\bar{x}$ is determined by the next two parameters.

lowbnd: This parameter can take any value between 0 and 100 and specifies the percentage of bounds present in the problem which are lower bounds. Once the number of lower bounds is fixed we determine randomly which are the corresponding indices. The same is done for the upper bounds if the bounds are not required to come in pairs.

actbnd: This parameter can vary, in principle, between 0 and 100 and specifies the percentage of existing bounds in the problem which are active at $\bar{x}$. The number of active lower bounds and the number of active upper bounds is proportional to the number of lower and upper bounds, respectively. As before, these bounds are randomly chosen. This parameter has to be selected with some caution because if, for

example, the number of bounds is greater than n and $actbnd = 100$ we are generating a problem with more than n variables at their bounds! However, this problem can never arise as long as $actbnd \leq 50$.

We make an example to illustrate the use of these three parameters. Suppose that $n = 1000$, $bnd = 50$, and $lowbnd = 40$. Then, there are 400 lower bounds and 600 upper bounds. If $actbnd = 50$ then 200 lower bounds and 300 upper bounds are active at $\bar{x}$, while $\bar{x}$ is an unconstrained minimum if $actbnd = 0$.

ndeg: This parameter can be any nonnegative number. It controls the degeneracy of the problem and is chosen in a way similar to that proposed in [31]. We have already noted that if we define h_i by (6.4) then β_i , for $i \in \bar{L} \cup \bar{U}$, are the multipliers associated with the active constraints. We have set

$$\beta_i = 10^{-ndeg \mu_i}, \quad i \in \bar{L} \cup \bar{U},$$

where μ_i is a random number uniformly distributed in $[0, 1]$. The larger *ndeg* is the more degenerate the problem is.

width: This parameter can take any positive value and controls the value of the bounds which are inactive. More precisely, if x_i has a finite inactive lower bound, we set $l_i = \bar{x}_i - width$, while we set $u_i = \bar{x}_i + width$ if x_i has a finite inactive upper bound.

To complete the description of the generation process, it only remains to specify the starting point x^0 . Our basic idea was to use a feasible point that was obtained by perturbing the standard unconstrained starting point y^0 of the underlying unconstrained problem. We have set

$$x_i^0 = \begin{cases} l_i + 5 & \text{if } y_i^0 \leq l_i, \\ y_i^0 & \text{if } l_i < y_i^0 < u_i, \\ u_i - 5 & \text{if } y_i^0 \geq u_i. \end{cases} \quad i = 1, \dots, n,$$

We have considered only $width \geq 5$ so that x^0 is always feasible.

In Tables 8–15 we report a first set of results on nondegenerate problems (*ndeg* = 0), that show the influence of the number and distribution of the bounds on the behavior of the algorithm.

We have generated test problems with an average number of constraints (Tables 8–10 with $bnd=50$), few constraints (Tables 11–13 with $bnd=10$), and many constraints (Tables 14–15 with $bnd=90$). For each case we consider problems having only lower bounds (Tables 8 and 11 with $lowbnd=100$),² an equal number of lower and upper bounds that are randomly distributed (Tables 9, 12, and 14 with $lowbnd=50$), and an equal number of lower and upper bounds that are randomly distributed, but in pairs, i.e., each variable either is not bounded or has both lower and upper bounds (Tables 10, 13, and 15). The meaning of the entries of these tables is the same as in the tables in the previous subsection. The algorithm always converged to the solution found by the unconstrained minimization.

These tables show that the number and distribution of the bounds do not seem to affect significantly the behavior of the algorithm, even if it should be noted that the minimal surface problem (4.3), appears to be more difficult than the others. This is because in some iterations, in correspondence to nearly singular projected Hessian matrices, overly large search directions d^k are produced and this slows down

²This choice is not possible when $bnd=90$ because this would imply 1800 lower bounds!

TABLE 8
 $bnd=50$, $lowbnd=100$, $actbnd=50$, $ndeg=0$, $width=10$.

Prob.	#it	#Bl	#f	80%	90%	100%	#L	#U	% at x^0
4.3	52	212	171	2	2	5	512	0	73.5
4.4	8	11	7	1	1	2	512	0	82.1
4.5	3	4	6	2	2	2	500	0	50.0
4.7	3	6	6	1	1	2	512	0	50.0

TABLE 9
 $bnd=50$, $lowbnd=50$, $actbnd=50$, $ndeg=0$, $width=10$.

Prob.	#it	#Bl	#f	80%	90%	100%	#L	#U	% at x^0
4.3	60	61	262	1	1	1	256	256	62.6
4.4	5	8	7	1	1	2	256	256	60.5
4.5	3	7	6	3	3	3	250	250	0.2
4.7	3	6	6	1	2	2	256	256	6.4

TABLE 10
 $bnd=50$, $lowbnd=50$, $actbnd=50$, $ndeg=0$, $width=10$, with bounds in pairs.

Prob.	#it	#Bl	#f	80%	90%	100%	#L	#U	% at x^0
4.3	70	205	244	1	1	1	256	256	72.4
4.4	5	8	7	1	1	2	256	256	71.6
4.5	4	10	7	3	4	4	250	250	35.6
4.7	3	6	6	1	1	2	256	256	38.5

TABLE 11
 $bnd=10$, $lowbnd=100$, $actbnd=50$, $ndeg=0$, $width=10$.

Prob.	#it	#Bl	#f	80%	90%	100%	#L	#U	% at x^0
4.3	21	30	87	0	1	1	102	0	89.2
4.4	14	17	9	0	0	2	102	0	95.1
4.5	3	7	6	1	2	3	100	0	50.0
4.7	2	3	6	1	1	1	102	0	50.0

TABLE 12
 $bnd=10$, $lowbnd=50$, $actbnd=50$, $ndeg=0$, $width=10$.

Prob.	#it	#Bl	#f	80%	90%	100%	#L	#U	% at x^0
4.3	95	96	449	1	1	1	51	51	63.7
4.4	13	15	8	1	1	2	51	51	57.8
4.5	3	5	6	3	3	3	50	50	0.0
4.7	3	6	6	1	1	2	51	51	2.9

TABLE 13
 $bnd=10$, $lowbnd=50$, $actbnd=50$, $ndeg=0$, $width=10$, with bounds in pairs.

Prob.	#it	#Bl	#f	80%	90%	100%	#L	#U	% at x^0
4.3	47	84	171	0	1	1	51	51	81.3
4.4	13	15	9	1	1	2	51	51	75.0
4.5	5	8	8	3	4	4	50	50	44.4
4.7	3	5	6	1	1	2	51	51	43.6

convergence and also explains the large number of functions evaluations. This is a typical problem of Newton's method.

We now examine the influence of the parameter $width$ and $ndeg$. We tried the algorithm on problems generated with the average values for the parameters bnd , $lowbnd$, and $actbnd$ used in Table 9. For the parameter $width$, which determines

TABLE 14
 $bnd= 90, lowbnd= 50, actbnd= 50, ndeg= 0, width= 10$.

Prob.	#it	#Bl	#f	80%	90%	100%	#L	#U	% at x^0
4.3	52	53	286	1	1	1	460	460	53.1
4.4	5	8	7	1	2	2	460	460	59.4
4.5	3	16	6	2	2	3	450	450	18.3
4.7	3	6	6	1	2	2	460	460	10.8

TABLE 15
 $bnd= 90, lowbnd= 50, actbnd= 50, ndeg= 0, width= 10$ with bounds in pairs.

Prob.	#it	#Bl	#f	80%	90%	100%	#L	#U	% at x^0
4.3	35	67	121	1	1	1	460	460	55.1
4.4	6	9	9	1	2	2	460	460	62.2
4.5	3	16	6	2	2	3	450	450	23.6
4.7	3	6	6	1	2	2	460	460	20.3

the values of nonactive bounds, we considered the value 100 and 500 and the results obtained are equal to those reported in Table 9. In Table 16, instead, we consider the influence of the degeneracy for $ndeg$ equal to 1, 3 and ∞ . This last case corresponds to completely degenerate problems, i.e., $\beta_i = 0$ for $i \in \bar{L} \cup \bar{U}$.

TABLE 16
Influence of degeneracy: $bnd= 50, lowbnd= 50, actbnd= 50, width= 10$.

Prob.	$ndeg$	#it	#Bl	#f	80%	90%	100%	#L	#U	% at x^0
4.3	1	49	85	231	1	32	32	256	256	57.2
4.3	3	37	109	192	30	35	36	256	256	43.0
4.3	∞	65	224	381	65	65	65	256	256	26.5
4.4	1	8	13	8	2	2	3	256	256	55.8
4.4	3	6	15	9	3	3	4	256	256	50.7
4.4	∞	7	24	10	7	7	7	256	256	47.8
4.5	1	3	8	6	3	3	3	250	250	0.2
4.5	3	4	9	8	3	4	4	250	250	1.6
4.5	∞	5	7	8	4	5	5	250	250	0.2
4.7	1	3	8	6	2	2	3	256	256	6.4
4.7	3	4	14	7	3	3	4	256	256	5.2
4.7	∞	7	14	10	7	7	7	256	256	23.4

The influence of degeneracy is fairly weak. An increase in the number of iterations can be detected as the degeneracy increases, while a larger number of iterations of the block pivoting algorithm is needed to solve the subproblems. This was expected, since on degenerate problems the identification technique of the ASNМ method is less efficient so that more work is needed by the subproblem solver. Degeneracy also reflects in the columns concerning the first iteration at which a given percentage of the constraints is correctly identified. We recall that in the case of $ndeg = \infty$, the multipliers are all 0. This implies that, even very close to the solution, every active bound can unpredictably be classified as active or not active by the guessing technique.

The starting point we have used in the runs described so far is reasonably far from the solution. However, it is of interest to test the algorithm also using still harder starting points. To this end we considered the following starting point, suggested by

one of the referees.

$$(7.3) \quad x_i^0 = \begin{cases} u_i & \text{if } y_i^0 \leq l_i \text{ and } u_i \text{ is finite,} \\ l_i + 5 & \text{if } y_i^0 \leq l_i \text{ and } u_i \text{ is infinite,} \\ y_0 & \text{if } l_i < y_i^0 < u_i \text{ and at least one bound is infinite,} \\ (l_i + u_i)/2 & \text{if } l_i < y_i^0 < u_i \text{ and both bounds are finite,} \\ l_i & \text{if } y_i^0 \geq u_i \text{ and } l_i \text{ is finite,} \\ u_i - 5 & \text{if } y_i^0 \geq u_i \text{ and } l_i \text{ is infinite.} \end{cases}$$

In Table 17 we summarize the result obtained for the settings of parameters considered in Tables 8–16. In this table the parameter p indicates whether the bounds are in pairs ($p = 1$) or not ($p = 0$) and *width* is always equal to 10. We see that there seems to be no big qualitative difference between these results and those presented in the previous analogous tables. The starting points generated according to (7.3) are usually farther away from the solution than the one considered previously and therefore an increase in the number of iterations has to be expected. However, the increase is not excessive. Problem 4.3 is still the hardest problem, and, at a closer exam, the difficulty always appeared to be due to an almost singular projected Hessian matrix. This is also confirmed by the fact that better results can be obtained by changing the parameter δ used in the modified Cholesky factorization (see section 5.1). This suggests that an adaptive choice of the parameter δ could probably lead to a better code, but we leave these kinds of issues to future research. In half of the runs with problem 4.7 the function value sequence diverged to $-\infty$. This is due to the known fact that, see [1], the objective function for this problem is unbounded below, and that the attraction region of the local minimizer is not large.

8. Conclusions. In this paper we have introduced a new active set Newton's algorithm for the solution of large-scale box constrained problems. The implementation of the algorithm uses sparse matrix techniques to estimate and store the Hessian of the objective function and to solve the required linear systems. The method can be seen as an intermediate approach between recursive quadratic programming (RQP) and active set algorithms. Each of these approaches has its advantages and drawbacks. If, as in RQP algorithms, we take into account all the constraints in the quadratic subproblems, we are able to find good directions whose calculation, however, may be expensive. Instead, if we use an active set strategy, the cost of computing the search direction is greatly reduced. However, the active set strategies generally employed only allow small changes in the active set from one iteration to the next, and this can slow down convergence in the large-scale case. In our method, a guessing strategy that allows rapid changes in the active set estimate is incorporated, while the direction corresponding to the free variables is calculated by solving a (reduced) quadratic subproblem that takes into account only the constraints corresponding to the free variables. So we try to retain the advantages of both methods, while mitigating their drawbacks.

The preliminary numerical results reported seem to indicate that our method is robust and fairly insensitive to the parameters that are generally considered when studying the behavior of algorithms for box constrained problems. Therefore, although the cost of each iteration may be higher than that of a truncated scheme or a limited memory quasi-Newton method, we feel that the algorithm proposed in this paper could be useful in the solution of general box constrained problems. In our view, the main drawback of the method is that it requires the user to provide information about the structure of the Hessian and this can be, sometimes, too difficult.

TABLE 17
Results for starting point (7.3) ($-\infty$ if an unbounded sequence is generated).

Pr.	bnd	lowbnd	actbnd	ndeg	p	#it	#Bl	#f	80%	90%	100%	% x^0
4.3	50	100	50	0	0	52	212	171	2	2	5	73.5
	50	50	50	0	0	73	131	345	1	1	67	0.0
	50	50	50	0	1	97	251	485	1	1	1	59.6
	10	100	50	0	0	21	30	87	0	1	1	89.2
	10	50	50	0	0	113	114	582	1	1	1	0.0
	10	50	50	0	1	29	53	108	1	1	1	68.0
	90	50	50	0	0	81	149	429	1	73	73	0.0
	90	50	50	0	1	45	106	169	1	1	38	18.3
	50	50	50	1	0	289	817	2037	252	252	252	0.0
	50	50	50	3	0	89	279	422	73	81	87	0.0
	50	50	50	∞	0	148	491	1029	148	148	148	31.0
4.4	50	100	50	0	0	8	11	7	1	1	2	82.1
	50	50	50	0	0	5	8	7	2	2	2	20.5
	50	50	50	0	1	7	10	8	1	2	2	55.7
	10	100	50	0	0	14	17	9	0	0	2	95.0
	10	50	50	0	0	7	10	8	2	2	2	9.8
	10	50	50	0	1	18	20	9	1	1	2	58.1
	90	50	50	0	0	4	7	7	2	2	2	34.5
	90	50	50	0	1	4	7	7	2	2	2	41.0
	50	50	50	1	0	6	12	8	2	3	3	19.1
	50	50	50	3	0	9	23	19	4	4	8	18.9
	50	50	50	∞	0	8	26	11	8	8	8	46.7
4.5	50	100	50	0	0	3	4	6	2	2	2	50.0
	50	50	50	0	0	3	9	6	3	3	3	11.4
	50	50	50	0	1	4	11	7	3	4	4	36.9
	10	100	50	0	0	3	7	6	1	2	3	50.0
	10	50	50	0	0	3	5	6	3	3	3	5.0
	10	50	50	0	1	5	8	8	3	4	4	44.4
	90	50	50	0	0	3	7	6	2	2	3	45.3
	90	50	50	0	1	3	10	6	2	2	3	43.5
	50	50	50	1	0	3	9	6	3	3	3	11.4
	50	50	50	3	0	4	10	9	3	3	4	12.8
	50	50	50	∞	0	4	8	7	4	4	4	11.4
4.7	50	100	50	0	0	3	6	6	1	1	2	50.0
	50	50	50	0	0	$-\infty$						
	50	50	50	0	1	$-\infty$						
	10	100	50	0	0	2	3	6	1	1	1	50.0
	10	50	50	0	0	$-\infty$						
	10	50	50	0	1	3	5	6	1	1	2	47.0
	90	50	50	0	0	4	8	7	2	2	2	15.5
	90	50	50	0	1	5	9	8	2	2	4	24.1
	50	50	50	1	0	$-\infty$						
	50	50	50	3	0	$-\infty$						
	50	50	50	∞	0	$-\infty$						

Furthermore, large-scale dense problems cannot be solved by this kind of approach. A solution to these problems could possibly come from the use of (sparse) quasi-Newton methods to update the matrix B^k , but we leave this point to future research.

There are several other issues that could lead to improvements. A first point that should be considered is probably the choice of the stabilization technique. We have adopted the nonmonotone scheme of [28], but this is not the only possible choice, and the framework developed in section 2 easily allows one to consider different techniques. Another important point that should be further investigated is the solution of the quadratic subproblem. On average the block pivoting strategy adopted seems efficient

and even when, in a few cases, many inner iterations are needed, the cost is not high because of the relatively small cost of a single inner iteration. However, one may ask whether there are better ways of solving the quadratic subproblems in our framework, or whether it is possible to design a truncated version of the algorithm. The choice of the search direction when the projected Hessian is not sufficiently positive definite is another important point. To this end, it is interesting to note that the routine MA27 is capable of solving indefinite systems and cheaply computing, at the same time, a direction of negative curvature. The use of this direction could be a good alternative to performing the modified Cholesky factorization.

Acknowledgment. We are grateful to two anonymous referees whose detailed and constructive criticisms greatly improved this paper.

REFERENCES

- [1] B. AVERICK, R. CARTER, J. MORÉ, AND G.-L. XUE, *The Minpack-2 Test Problem Collection*, Tech. report MCS-P153-0692, Argonne National Laboratory, Mathematics and Computer Science Division, Argonne, IL, June 1992.
- [2] D. BERTSEKAS, *On the Goldstein-Levitin-Polyak gradient projection method*, IEEE Trans. Automat. Control, AC-21 (1976), pp. 174–184.
- [3] D. BERTSEKAS, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.
- [4] D. BERTSEKAS, *Projected Newton methods for optimization problems with simple constraints*, SIAM J. Control Optim., 20 (1982), pp. 221–246.
- [5] I. BONGARTZ, A. CONN, N. GOULD, AND P. TOINT, *CUTE: Constrained and unconstrained testing environment*, ACM Trans. Math. Software, 21 (1995), pp. 123–160.
- [6] J. BURKE AND J. MORÉ, *On the identification of active constraints*, SIAM J. Numer. Anal., 25 (1988), pp. 1197–1211.
- [7] J. BURKE AND J. MORÉ, *Exposing constraints*, SIAM J. Optim., 4 (1994), pp. 573–595.
- [8] P. CALAMAI AND J. MORÉ, *Projected gradient methods for linearly constrained problems*, Math. Programming, Ser. B, 39 (1987), pp. 93–116.
- [9] R. CARTER, *Fast Numerical Determination of Symmetric Sparsity Patterns*, Tech. report MCS-P326-0992, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1992.
- [10] T. COLEMAN AND L. HULBERT, *A direct active set algorithm for large sparse quadratic programs with simple bounds*, Math. Programming, Ser. B, 20 (1983), pp. 187–209.
- [11] T. COLEMAN AND J. MORÉ, *Estimation of sparse Jacobians matrices and graph coloring problems*, SIAM J. Numer. Anal., 10 (1984), pp. 346–347.
- [12] A. CONN, N. GOULD, AND P. TOINT, *Global convergence of a class of trust region algorithms for optimization with simple bounds*, SIAM J. Numer. Anal., 25 (1988), pp. 433–460.
- [13] A. CONN, N. GOULD, AND P. TOINT, *Testing a class of methods for solving minimization problems with simple bounds on the variables*, Math. Comp., 50 (1988), pp. 399–430.
- [14] A. CONN, N. GOULD, AND P. TOINT, *LANCELOT- A Fortran Package for Large-Scale Nonlinear Optimization*, Springer-Verlag, Berlin, New York, 1992.
- [15] R. COTTLE AND M. GOHEEN, *A special class of large quadratic programs*, in Nonlinear Programming 3, R. M. Meyer, O. Mangasarian, and S. Robinson, eds., Academic Press, New York, 1978, pp. 361–390.
- [16] J. DANIEL, *Stability of the solution of definite quadratic programs*, Math. Programming, 5 (1973), pp. 41–53.
- [17] G. DI PILLO, F. FACCHINEI, AND L. GRIPPO, *An RQP algorithm using a differentiable exact penalty function for inequality constrained problems*, Math. Programming, 55 (1992), pp. 49–68.
- [18] G. DI PILLO AND L. GRIPPO, *A class of continuously differentiable exact penalty function algorithms for nonlinear programming problems*, in System Modelling and Optimization, P. Toft-Christensen, ed., Springer-Verlag, Berlin, 1984, pp. 246–256.
- [19] G. DI PILLO, L. GRIPPO, AND S. LUCIDI, *Globally convergent exact penalty algorithms for constrained optimization*, in System Modelling and Optimization, A. Prekopa, J. Szelezsan, and B. Strazicky, eds., Springer-Verlag, Berlin, 1986, pp. 694–703.
- [20] I. DUFF, A. ERISMAN, AND J. REID, *Direct Methods for Sparse Matrices*, Clarendon Press, Oxford, 1986.

- [21] I. DUFF AND J. REID, *The multifrontal solution of indefinite sparse symmetric linear equations*, ACM Trans. Math. Software, 9 (1984), pp. 302–325.
- [22] J. DUNN, *Global and asymptotic convergence rate estimates for a class of projected gradient process*, SIAM J. Control Optim., 19 (1981), pp. 368–400.
- [23] F. FACCHINEI, *Minimization of SC^1 functions and the Maratos effect*, Oper. Res. Lett., 17 (1995), pp. 131–137.
- [24] F. FACCHINEI AND S. LUCIDI, *A Class of Penalty Functions for Optimization Problems with Bound Constraints. Part 2: Algorithms and Numerical Results*, Tech. report 336, IASI-CNR, Roma, Italy, 1992.
- [25] F. FACCHINEI AND S. LUCIDI, *Quadratically and superlinearly convergent algorithms for the solution of inequality constrained minimization problems*, J. Optim. Theory Appl., 85 (1995), pp. 265–289.
- [26] P. GILL AND W. MURRAY, *Newton-type methods for unconstrained and linearly constrained optimization*, Math. Programming, 28 (1974), pp. 311–350.
- [27] P. GILL, W. MURRAY, AND M. WRIGHT, *Practical Optimization*, Academic Press, New York, 1981.
- [28] L. GRIPPO, F. LAMPARIELLO, AND S. LUCIDI, *A class of nonmonotone stabilization methods in unconstrained optimization*, Numer. Math., 59 (1991), pp. 779–805.
- [29] J. JÚDICE AND F. PIRES, *A block principal pivoting algorithm for large-scale strictly monotone linear complementarity problems*, Comput. Oper. Res., 21 (1994), pp. 587–596.
- [30] M. LESCRENIER, *Convergence of trust region algorithms for optimization with bounds when strict complementarity does not hold*, SIAM J. Numer. Anal., 28 (1991), pp. 476–495.
- [31] J. MORÉ AND G. TORALDO, *Algorithms for bound constrained quadratic programming problems*, Numer. Math., 55 (1989), pp. 377–400.
- [32] J. MORÉ AND G. TORALDO, *Numerical solution of large quadratic programming problems with bound constraints*, SIAM J. Control Optim., 1 (1991), pp. 93–113.
- [33] S. NASH AND A. SOFER, *A barrier method for large-scale constrained optimization*, ORSA J. Comput., 5 (1993), pp. 40–53.
- [34] J. ORTEGA AND W. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.
- [35] M. POWELL AND P. TOINT, *On the estimation of sparse hessian matrices*, SIAM J. Numer. Anal., 16 (1979), pp. 1060–1074.
- [36] R. SCHNABEL AND E. ESKOW, *A new modified Cholesky factorization*, SIAM J. Sci. Statist. Comput., 11 (1990), pp. 1136–1158.
- [37] J. SOARES, *Métodos de Newton para Optimizacão com Restrições Simples*, Master's thesis, Departamento de Matemática, Universidade de Coimbra, Coimbra, Portugal, 1993.
- [38] J. SOARES, J. JÚDICE, AND F. FACCHINEI, *An Active Set Newton's Methods for Large-Scale Nonlinear Programs with Box Constraints*, Tech. report 367, IASI-CNR, Roma, Italy, 1993.

ON THE CONVERGENCE OF CONSTRAINED PARALLEL VARIABLE DISTRIBUTION ALGORITHMS*

MICHAEL V. SOLODOV†

Abstract. We consider the parallel variable distribution (PVD) approach proposed by Ferris and Mangasarian [*SIAM J. Optim.*, 4 (1994), pp. 815–832] for solving optimization problems. The problem variables are distributed among p processors with each processor having the primary responsibility for updating its block of variables while allowing the remaining “secondary” variables to change in a restricted fashion along some easily computable directions. For constrained nonlinear programs, convergence in [M. C. Ferris and O. L. Mangasarian, *SIAM J. Optim.*, 4 (1994), pp. 815–832] was established in the special case of convex block-separable constraints. For general (inseparable) constraints, it was suggested that a dual differentiable exact penalty function reformulation of the problem be used. We propose to apply the PVD approach to problems with general convex constraints directly and show that the algorithm converges, provided certain conditions are imposed on the change of secondary variables. These conditions are both natural and practically implementable. We also show that the original requirement of exact global solution of the parallel subproblems can be replaced by a less stringent sufficient descent condition. The first rate of convergence result for the class of constrained PVD algorithms is also given.

Key words. parallel optimization, nonlinear programming, linear convergence

AMS subject classifications. 90C30, 49D27

PII. S1052623495293949

1. Introduction. We consider the general nonlinear programming problem

$$(1.1) \quad \min_{x \in C} f(x),$$

where C is a nonempty closed convex set in $\mathbb{R}^n$ and $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a continuously differentiable function. We first describe the PVD algorithm proposed in [4] and, in the unconstrained case, further studied and extended in [25, 7]. Let the problem variables $x \in \mathbb{R}^n$ be partitioned into p blocks $x_1, \dots, x_p$ such that $x_l \in \mathbb{R}^{n_l}$, $\sum_{l=1}^p n_l = n$. These blocks of variables are then distributed among p parallel processors. Each processor has the primary responsibility for updating its block of variables by solving the parallelization problem (see Algorithm 1.1 below). The remaining “secondary” variables are allowed to change in a restricted fashion along some easily computable directions. The distinctive novel feature of this algorithm is the presence of the “forget-me-not” term $x_l^i + D_l^i \mu_l$ in the parallel subproblems (1.2) which allows for a change in “secondary” variables. This makes PVD fundamentally different from the block Jacobi [2], coordinate descent [26], and parallel gradient distribution algorithms [16]. The forget-me-not approach improves robustness and accelerates convergence of the algorithm and is the key to its success. The parallelization phase in Algorithm 1.1 is followed by a simple synchronization step which selects a point with the objective function value at least as good as the smallest among all the new points computed by the parallel processors.

We now formally state the algorithm.

*Received by the editors October 27, 1995; accepted for publication (in revised form) November 13, 1996. This material is based on research supported by CNPq grant 300734/95-6.

<http://www.siam.org/journals/siopt/8-1/29394.html>

†Instituto de Matemática Pura e Aplicada, Estrada Dona Castorina 110, Jardim Botânico, Rio de Janeiro, RJ 22460-320, Brazil (solodov@impa.br).

ALGORITHM 1.1 (PVD). *Start with any $x^0 \in C$. Having x^i , check a stopping criterion (to be specified later). If it is not satisfied, compute x^{i+1} as follows:*

- **Parallelization:** *For each processor $l \in \{1, \dots, p\}$ compute*

$$(1.2) \quad (y_l^i, \mu_{\bar{l}}^i) \in \arg \min_{x_l, \mu_{\bar{l}}} \psi_l^i(x_l, \mu_{\bar{l}}) := f(x_l, x_{\bar{l}}^i + D_{\bar{l}}^i \mu_{\bar{l}})$$

subject to $(x_l, x_{\bar{l}}^i + D_{\bar{l}}^i \mu_{\bar{l}}) \in C$.

- **Synchronization:** *Compute x^{i+1} such that*

$$(1.3) \quad f(x^{i+1}) \leq \min_{l \in \{1, \dots, p\}} \psi_l^i(y_l^i, \mu_{\bar{l}}^i).$$

In the above algorithm $\bar{l}$ denotes the complement of l in the set $\{1, \dots, p\}$ and $\mu_{\bar{l}} \in \mathbb{R}^{p-1}$. The matrix $D_{\bar{l}}^i$ is an $n_{\bar{l}} \times (p-1)$ block-diagonal matrix formed by placing the blocks $d_1^i, \dots, d_{p-1}^i$ ($d_t^i \in \mathbb{R}^{n_t}$, $t = 1, \dots, p-1$) of an arbitrary direction $d^i \in \mathbb{R}^n$ (e.g., quasi-Newton or steepest descent directions in the case $C = \mathbb{R}^n$) along its block diagonal as follows:

$$D_{\bar{l}}^i := \begin{pmatrix} d_1^i & & & & & \\ & d_2^i & & & & \\ & & \ddots & & & \\ & & & d_{l-1}^i & & \\ & & & & d_{l+1}^i & \\ & & & & & \ddots \\ & & & & & & d_p^i \end{pmatrix}.$$

The synchronization step in Algorithm 1.1 may consist of minimizing the objective function in the affine hull, subject to feasibility, of all the points computed in parallel by the p processors. In principle, any point with the objective function value at least as good as the smallest computed by all the processors is acceptable. We refer the reader to [4] for a study of computational efficiency of the PVD approach and other practical issues.

In [4] it was shown that in the special case when the feasible set C has block-separable structure (i.e., C is a Cartesian product of closed convex sets), every accumulation point of the PVD iterates satisfies the minimum principle necessary optimality condition [15]. It was further stated that in the case of inseparable constraints, the PVD approach may fail. The following example was employed to illustrate this conclusion.

$$\begin{aligned} \min_{x_1, x_2} f(x) &:= x_1^2 + x_2^2 \\ \text{subject to} \quad &0 \leq g(x) := x_1 + x_2 - 2. \end{aligned}$$

This strongly convex program has a unique global solution at $x_1 = x_2 = 1$. However, the point $x_1 = 0.5$, $x_2 = 1.5$ is a global minimum with respect to both x_1 and x_2 , separately, subject to the problem constraint

$$\begin{aligned} 0.5 &= \arg \min_{x_1} \{x_1^2 + 2.25 \mid x_1 \geq 0.5\}, \\ 1.5 &= \arg \min_{x_2} \{0.25 + x_2^2 \mid x_2 \geq 1.5\}. \end{aligned}$$

Therefore, if we use the above point as x^0 in Algorithm 1.1 and we set $D_1^0 = D_2^0 = 0$, then the algorithm will fail to make any progress towards solving the original problem. Moreover, every point in the set $\{x \mid g(x) = 0, x \geq 0\}$ has the same property, while only one of them is the true solution of the original problem. It was therefore argued that having a stationary point that results from minimizing the objective function with respect to individual blocks of variables and subject to the problem constraints does not result in a useful point unless the constraints are separable [4]. While this is true for an arbitrary choice of directions in D_l^i (note that in the example above, zero directions were used), it seems clear that one of the keys to success of any PVD algorithm should be precisely a reasonable choice of directions for change in the secondary variables. In fact, the presence of forget-me-not terms is what separates PVD from other related parallel techniques [2]. It is therefore natural to choose the directions in a special way and study their effect on the properties of the algorithm. In [25], by imposing natural restrictions on these directions, we were able to strengthen convergence results for unconstrained PVD. A reasonable choice of directions also allowed us to propose useful generalizations, such as algorithms with inexact subproblem solution and a certain degree of asynchronization [25].

We now define the following *projected gradient* residual function

$$(1.4) \quad r(x) := x - P_C[x - \nabla f(x)],$$

where $P_C[\cdot]$ stands for the orthogonal projection map onto the closed convex set C ; i.e., $P_C[y] := \arg \min_{x \in C} \|x - y\|$. It is well known that some $x \in \mathbb{R}^n$ satisfies the minimum principle necessary optimality condition [15] for problem (1.1)

$$x \in C, \quad \langle \nabla f(x), y - x \rangle \geq 0 \quad \forall y \in C$$

if and only if $r(x) = 0$. We shall call such x a stationary point of (1.1). Our stopping criterion for Algorithm 1.1 will consist of checking whether at the current iterate x^i the minimum principle optimality condition is satisfied; i.e., $r(x^i) = 0$.

In this paper, we propose to employ the residual (projected gradient) function (1.4) for the change of secondary variables in problems with general convex constraints. In particular, we set $d^i := r(x^i)$. Then the matrices D_l^i are comprised of the portions of the projected gradient direction, that is

$$(1.5) \quad d_l^i = r_l(x^i) := [r(x^i)]_l, \quad l = 1, \dots, p,$$

where $[r(x^i)]_l \in \mathbb{R}^{n_l}$ is the portion of $r(x^i)$ corresponding to $x_l \in \mathbb{R}^{n_l}$. Let us go back to the example above. As before, consider the point $x_1^0 = 0.5$, $x_2^0 = 1.5$. It is easy to check that $r_1(x^0) = -1$ and $r_2(x^0) = 1$. Thus the two parallel subproblems in the PVD algorithm are

$$\begin{aligned} \min_{x_1, \mu_2} f(x_1, x_2^0 + \mu_2 r_2(x^0)) &= (x_1)^2 + (\mu_2)^2 + 2x_2^0 \mu_2 + (x_2^0)^2 \\ \text{subject to} \quad 0 &\leq g(x_1, x_2^0 + \mu_2 r_2(x^0)) = x_1 + \mu_2 - 0.5 \end{aligned}$$

and

$$\begin{aligned} \min_{x_2, \mu_1} f(x_1^0 + \mu_1 r_1(x^0), x_2) &= -2x_1^0 \mu_1 + (\mu_1)^2 + (x_2)^2 + (x_1^0)^2 \\ \text{subject to} \quad 0 &\leq g(x_1^0 + \mu_1 r_1(x^0), x_2) = x_2 - \mu_1 - 1.5. \end{aligned}$$

The solution of the first subproblem is $y_1^0 = 1$, $\mu_2^0 = -0.5$, and the solution of the second is $y_2^0 = 1$, $\mu_1^0 = -0.5$. It is easy to see that $x_1^0 + \mu_1^0 r_1(x^0) = 1$ and $x_2^0 + \mu_2^0 r_2(x^0) = 1$. So both subproblems produce the solution of the original problem.

In section 2 we show that, for the choice of directions given by (1.5), the PVD algorithm converges under no special assumptions on the structure of the feasible set (see Theorem 2.1). Moreover, we prove convergence for the case of Hölder continuous gradient of the objective function (Lipschitz continuity was assumed in [4] and [25]). We are further able to give the first rate of convergence result for the constrained PVD methods (Theorem 2.2). Finally, in section 2 we also show that the exact global solution requirement for parallel subproblems (1.2) can be replaced by a less stringent condition of sufficient descent (see (2.3)). This yields a more practical approach and makes it easier to achieve good load balancing among the parallel processors. Some concluding remarks are given in section 3.

We briefly describe our notation now. The usual inner product of two vectors $x \in \mathbb{R}^n$, $y \in \mathbb{R}^n$ is denoted by $\langle x, y \rangle$. The Euclidean 2-norm of $x \in \mathbb{R}^n$ is given by $\|x\|^2 = \langle x, x \rangle$. For a nonempty (closed) set $S \subset \mathbb{R}^n$, $d(\cdot, S)$ denotes the Euclidean distance to the set S . For a differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, ∇f will denote the n -dimensional vector of partial derivatives with respect to x . If a function $f(\cdot)$ has Hölder continuous partial derivatives on $\mathbb{R}^n$ with some constant $L > 0$ and modulus $\alpha \in (0, 1]$, that is

$$\|\nabla f(y) - \nabla f(x)\| \leq L\|y - x\|^\alpha \quad \forall x, y \in \mathbb{R}^n,$$

we write $f(\cdot) \in C_L^{1,\alpha}(\mathbb{R}^n)$. By R -linear convergence and Q -linear convergence, we mean linear convergence in the root sense and in the quotient sense, respectively, as defined in [19].

2. Convergence of constrained PVD algorithms. In this section we show that, in the general convex-constrained case, for the PVD approach to be effective it is crucial to impose certain conditions on the change of secondary variables. Fortunately, those conditions are natural and easily implementable in practice. The importance of change in the secondary variables was mentioned and experimentally established in [4]. However, convergence analysis in [4] fails to take advantage of forget-me-not terms. This inevitably weakens the properties of PVD algorithms, especially in the constrained case. In fact, it was suggested that the only sensible way to distribute variables for problems with inseparable constraints is to convert them to unconstrained problems. This can be done in several ways, but, unfortunately, each has some disadvantages. The exterior penalty approach [5] requires the unboundness of the penalty parameter, while the augmented Lagrangian methods [24, 1] change the minimization problem into a saddle-point problem which is, in general, more difficult. In [4] it was suggested to use a certain dual differentiable exact penalty function formulation [8] for which the penalty parameter remains finite. This is a reasonable approach, but it also has some drawbacks. In particular, by using this dual penalty function formulation we increase the dimensionality of the problem and may lose any special structure if it was present in the problem. We propose an alternative approach in which we deal directly with the original problem (1.1).

Theorem 2.1 below shows that PVD methods can be successfully applied to *any* optimization problem with general convex constraints. Furthermore, considering specific directions and making explicit use of their properties in our analysis, we are able to obtain stronger convergence results, including rate of convergence results. This

was also the case for the analysis of unconstrained PVD algorithms [25]. Another improvement consists of extending the analysis to functions in the class $C_L^{1,\alpha}(\mathbb{R}^n)$ where $\alpha \in (0, 1]$ (in [4, 25] the case of $\alpha = 1$ is considered).

We are now ready to prove our main results.

THEOREM 2.1. *Let $f(\cdot) \in C_L^{1,\alpha}(\mathbb{R}^n)$. Suppose $\{x^i\}$ is any sequence generated by PVD Algorithm 1.1 with the choice of directions for secondary variables given by (1.5). Then either $f(\cdot)$ is unbounded from below on C or the sequence $\{f(x^i)\}$ converges, the sequence $\{r(x^i)\}$ converges to zero and every accumulation point of the sequence $\{x^i\}$ satisfies the minimum principle necessary optimality condition.*

Proof. Take any $\beta \in (0, 1)$. We first show that for any iteration $i = 0, 1, \dots$ and any processor $l = 1, \dots, p$, the point

$$(x_l^i - \eta_i r_l(x^i), -\eta_i e_{\bar{l}}) \in \mathbb{R}^{n_l + p - 1}$$

is feasible for the corresponding subproblem (1.2) of minimizing the function $\psi_l^i(x_l, \mu_{\bar{l}})$. In the above, $e_{\bar{l}}$ is a vector of ones of appropriate dimension, and

$$\eta_i := \min \left\{ 1, \beta \left(\frac{1 + \alpha}{L} \right)^{1/\alpha} \|r(x^i)\|^{(1-\alpha)/\alpha} \right\}.$$

Indeed,

$$\begin{aligned} (x_l^i - \eta_i r_l(x^i), x_{\bar{l}}^i - \eta_i D_{\bar{l}}^i e_{\bar{l}}) &= (x_l^i - \eta_i r_l(x^i), x_{\bar{l}}^i - \eta_i r_{\bar{l}}(x^i)) \\ &= x^i - \eta_i r(x^i) \\ &= (1 - \eta_i) x^i + \eta_i P_C[x^i - \nabla f(x^i)] \in C, \end{aligned}$$

where the first equality follows from the block diagonal structure of $D_{\bar{l}}^i$ and (1.5), the last equality follows from (1.4), and the inclusion is by convexity of the set C . We further obtain

$$\begin{aligned} f(x^i) - f(y_l^i, x_{\bar{l}}^i + D_{\bar{l}}^i \mu_{\bar{l}}^i) &= \psi_l^i(x_l^i, 0) - \psi_l^i(y_l^i, \mu_{\bar{l}}^i) \\ &\geq \psi_l^i(x_l^i, 0) - \psi_l^i(x_l^i - \eta_i r_l(x^i), -\eta_i e_{\bar{l}}) \\ &= f(x^i) - f(x_l^i - \eta_i r_l(x^i), x_{\bar{l}}^i - \eta_i D_{\bar{l}}^i e_{\bar{l}}) \\ &= f(x^i) - f(x^i - \eta_i r(x^i)), \end{aligned}$$

where the inequality follows from (1.2). Furthermore,

$$\begin{aligned} f(x^i - \eta_i r(x^i)) - f(x^i) &= - \int_0^{\eta_i} \langle \nabla f(x^i - tr(x^i)), r(x^i) \rangle dt \\ &= -\eta_i \langle \nabla f(x^i), r(x^i) \rangle \\ &\quad - \int_0^{\eta_i} \langle \nabla f(x^i - tr(x^i)) - \nabla f(x^i), r(x^i) \rangle dt \\ &\leq -\eta_i \langle \nabla f(x^i), r(x^i) \rangle + L \int_0^{\eta_i} t^\alpha \|r(x^i)\|^{1+\alpha} dt \\ &= -\eta_i \langle \nabla f(x^i), r(x^i) \rangle + \frac{L \eta_i^{1+\alpha}}{1 + \alpha} \|r(x^i)\|^{1+\alpha}, \end{aligned}$$

where the inequality follows from the Hölder continuity of $\nabla f(\cdot)$ and the Cauchy–Schwarz inequality. Hence

$$(2.1) \quad f(x^i) - f(y_l^i, x_{\bar{l}}^i + D_{\bar{l}}^i \mu_{\bar{l}}^i) \geq \eta_i \langle \nabla f(x^i), r(x^i) \rangle - \frac{L \eta_i^{1+\alpha}}{1 + \alpha} \|r(x^i)\|^{1+\alpha}.$$

By properties of the projection operator [22, p. 121], for any $x \in \mathbb{R}^n$ and any $y \in C$ it holds that

$$\langle x - P_C[x], y - P_C[x] \rangle \leq 0.$$

Taking $x = x^i - \nabla f(x^i)$ and $y = x^i \in C$, we have

$$\begin{aligned} 0 &\geq \langle x^i - \nabla f(x^i) - P_C[x^i - \nabla f(x^i)], x^i - P_C[x^i - \nabla f(x^i)] \rangle \\ &= \langle r(x^i) - \nabla f(x^i), r(x^i) \rangle. \end{aligned}$$

Hence

$$\langle \nabla f(x^i), r(x^i) \rangle \geq \|r(x^i)\|^2.$$

The latter relation combined with (2.1) yields

$$\begin{aligned} f(x^i) - f(y_l^i, x_l^i + D_l^i \mu_l^i) &\geq \eta_i \|r(x^i)\|^2 - \frac{L\eta_i^{1+\alpha}}{1+\alpha} \|r(x^i)\|^{1+\alpha} \\ &= \eta_i \|r(x^i)\|^2 \left(1 - \frac{L\eta_i^\alpha}{1+\alpha} \|r(x^i)\|^{\alpha-1} \right) \\ &\geq (1 - \beta^\alpha) \eta_i \|r(x^i)\|^2 \\ &\geq (1 - \beta^\alpha) \min \left\{ \|r(x^i)\|^2, \beta \left(\frac{1+\alpha}{L} \right)^{1/\alpha} \|r(x^i)\|^{(1+\alpha)/\alpha} \right\}, \end{aligned}$$

where the last two inequalities follow from the choice of η_i . By the synchronization step (1.3), we have

$$(2.2) \quad f(x^i) - f(x^{i+1}) \geq (1 - \beta^\alpha) \min \left\{ \|r(x^i)\|^2, \beta \left(\frac{1+\alpha}{L} \right)^{1/\alpha} \|r(x^i)\|^{(1+\alpha)/\alpha} \right\}.$$

Thus the sequence $\{f(x^i)\}$ is nonincreasing. If $f(\cdot)$ is bounded below on C , then $\{f(x^i)\}$ is bounded and hence it converges. In the latter case, $\{f(x^i) - f(x^{i+1})\} \rightarrow 0$ and therefore $\{r(x^i)\} \rightarrow 0$ by (2.2). By continuity of $r(\cdot)$, it follows that for every accumulation point $\bar{x}$ of the sequence $\{x^i\}$, $r(\bar{x}) = 0$. Thus all accumulation points of $\{x^i\}$ satisfy the minimum principle necessary optimality condition. $\square$

Some remarks are in order on the practical issues related to computing the projected gradient directions. When C is a polyhedral set (i.e., the constraints are linear), computing these directions requires solving a single quadratic programming problem at every synchronization step of Algorithm 1.1. For this, a wealth of fast and reliable algorithms is available [9]. In the case of nonlinear convex constraints, the task of computing the projected gradient directions is considerably more difficult. One of the referees for this paper made an interesting suggestion of computing the secondary directions by projecting onto the linearization of a nonlinear constraint set, much in the spirit of the iterative quadratic programming approach [6]. At this time, this is an open question as we do not have a proof verifying that this approach will work. However, it certainly deserves further investigation.

Careful reexamination of the proof of Theorem 2.1 shows that there is no need to compute exact global solutions for parallel subproblems (1.2). In particular, it is clear that the analysis is still valid if we accept any point (y_l^i, μ_l^i) that yields at least as good a descent for the function $\psi_l^i(\cdot, \cdot)$ as the point $(x_l^i - \eta_i r_l(x^i), -\eta_i e_l)$. We

can therefore replace the minimization problem in (1.2) with the following *sufficient descent* condition:

$$(2.3) \quad \text{Compute } (y_l^i, \mu_l^i) \text{ such that } \psi_l^i(y_l^i, \mu_l^i) \leq \psi_l^i(x_l^i - \eta_i r_l(x^i), -\eta_i e_l).$$

This observation is significant for several reasons. First of all, the global solution requirement is impractical if the objective function is not convex. And in any case, insisting on exact subproblem solution can be undesirable because it is likely to result in considerable idle times for processors that have already completed their work. It can be especially wasteful on the initial stages of minimization process. The sufficient descent criterion above shows that we are allowed a lot of flexibility in devising PVD algorithms. In particular, we can allow each of the p -parallel processors to take as many steps as desired. Synchronization can be performed at any time provided every processor has achieved the sufficient descent condition. This provides a flexible framework for effective load balancing. Finally, we note that (2.3) can be easily satisfied in practice if we take $(x_l^i, 0)$ as starting points when minimizing $\psi_l^i(\cdot, \cdot)$, $l = 1, \dots, p$, in subproblems (1.2).

In [4], the following *optimality function* was used to monitor the progress of the algorithm in the case of block-separable constraints:

$$\varphi(x) := -\min\{\langle \nabla f(x), h \rangle \mid x + h \in C, \|h\| \leq 1\}.$$

Then convergence results similar to our Theorem 2.1 follow from the relation

$$f(x^i) - f(x^{i+1}) \geq c \varphi(x^i), \quad c > 0,$$

and the fact that $\varphi(x) = 0$ if x satisfies the minimum principle necessary optimality condition and $\varphi(x) > 0$ otherwise. The use of a different optimality function

$$\varphi(x) := \|r(x)\|^2$$

allows us furthermore to give a rate of convergence result under the conditions similar to those in [13].

THEOREM 2.2. *Let $f(\cdot) \in C_L^{1,1}(\mathbb{R}^n)$ and $\{x^i\}$ be any sequence generated by Algorithm 1.1. Let accumulation points of the sequence $\{x^i\}$ exist and belong to the set $S := \arg \min_{x \in C} f(x) \neq \emptyset$. Suppose there exist positive constants c_j , $j = 2, \dots, 5$, such that*

$$(2.4) \quad \|r(x)\| \geq c_2 d(x, S) \quad \text{for all } x \text{ with } \|r(x)\| \leq c_3$$

and

$$(2.5) \quad f(x) - \min_{x \in C} f(x) \leq c_4 d(x, S)^2 \quad \text{for all } x \text{ with } d(x, S) \leq c_5.$$

Then the sequence $\{f(x^i)\}$ converges Q -linearly and the sequences $\{r(x^i)\}$ and $\{d(x^i, S)\}$ converge to zero R -linearly. If, furthermore, the solution set is a singleton ($S = \{\bar{x}\}$), then the sequence $\{x^i\}$ converges to $\bar{x}$ at least R -linearly.

Proof. By Theorem 2.1, $\{r(x^i)\} \rightarrow 0$ and hence also $\{d(x^i, S)\} \rightarrow 0$. It follows that relations (2.4) and (2.5) are satisfied for i sufficiently large, say $i \geq i_1$. Then, for $i \geq i_1$, from (2.2) with $\alpha = 1$ and (2.4), (2.5) we obtain

$$\begin{aligned} f(x^i) - f(x^{i+1}) &\geq c_1 c_2^2 d(x^i, S)^2 \\ &\geq \frac{c_1 c_2^2}{c_4} (f(x^i) - f^*), \end{aligned}$$

where $f^* = \min_{x \in C} f(x)$. Rearranging terms gives

$$f(x^{i+1}) - f^* \leq \left(1 - \frac{c_1 c_2^2}{c_4}\right) (f(x^i) - f^*).$$

Hence the sequence $\{f(x^i)\}$ converges Q -linearly. By (2.2), the sequence $\{r(x^i)\}$ converges to zero R -linearly and, by (2.4), $\{d(x^i, S)\}$ also converges to zero R -linearly.

If S is a singleton, that is $S = \{\bar{x}\}$, then $d(x^i, S) = \|x^i - \bar{x}\|$ and hence $\{x^i\}$ converges to $\bar{x}$ R -linearly. $\square$

Remark 2.1. It is an open question whether the sequence $\{x^i\}$ itself converges linearly under the assumptions of Theorem 2.2 when S is not a singleton. Note that if we had a serial algorithm where

$$x^{i+1} - x^i = -\eta_i r(x^i)$$

with the sequence of stepsizes $\{\eta_i\}$ uniformly bounded away from zero, then the linear convergence rate of $\{x^{i+1} - x^i\}$ (and hence also of $\{x^i\}$) would immediately follow from the linear convergence to zero of $\{r(x^i)\}$. The difficulty with the parallel algorithm is that we cannot explicitly relate $\{r(x^i)\}$ to $\{x^{i+1} - x^i\}$.

Condition (2.4) is sometimes referred to as a *projection-type error bound* [13, 14]. This error bound is known to hold when $f(\cdot)$ is quadratic and C is polyhedral [12, 23] or when $f(\cdot)$ is strongly convex [20] or when $f(\cdot)$ is a certain convex (not necessarily strictly convex) function and C is polyhedral [13] (see [14, Theorem 2.1] for a summary). Moreover, in certain situations, this condition holds globally (with $c_3 = \infty$) [10, 11, 17, 20]. Condition (2.5) states that $f(\cdot)$ does not grow faster than quadratically near its set of minimizers.

In conclusion, we note that the implicit Lagrangian reformulation [18] of the non-linear complementarity problem (NCP) satisfies (2.4)–(2.5) under certain assumptions. Consider the following NCP [3, 21] of finding an $x \in \mathbb{R}^n$ such that

$$F(x) \geq 0, \quad x \geq 0, \quad \langle x, F(x) \rangle = 0,$$

where $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$. In [18] it was established that the NCP can be solved via minimization of the following implicit Lagrangian function:

$$M(x, \alpha) := 2\alpha \langle x, F(x) \rangle + \| [x - \alpha F(x)]^+ \|^2 - \|x\|^2 + \| [F(x) - \alpha x]^+ \|^2 - \|F(x)\|^2,$$

where $\alpha > 1$ is a parameter and $[\cdot]^+$ denotes the orthogonal projection onto the nonnegative orthant $\mathbb{R}_+^n$. In particular, the implicit Lagrangian is nonnegative everywhere in $\mathbb{R}^n$ and assumes the value of zero precisely at the solutions of the NCP. One of the advantages of this reformulation is that if $F(\cdot)$ is continuously differentiable then so is $M(\cdot, \alpha)$.

First note that $\nabla M(\cdot, \alpha)$ is Lipschitz continuous on any bounded set if $F(\cdot)$ and $\nabla F(\cdot)$ are Lipschitz continuous and it is Lipschitz continuous everywhere if $F(\cdot)$ is affine. In [11] it was established that for all $x \in \mathbb{R}^n$ (and any $F(\cdot)$)

$$(2.6) \quad 2(\alpha - 1) \|x - [x - F(x)]^+\|^2 \leq M(x, \alpha) \leq 2\alpha(\alpha - 1) \|x - [x - F(x)]^+\|^2.$$

Let $S := \{x \in \mathbb{R}^n \mid x - [x - F(x)]^+ = 0\}$ be the (nonempty) solution set of the NCP. It can be easily seen that (2.5) is satisfied if the mapping $F(\cdot)$ is Lipschitz continuous

(with some modulus $\mu > 0$):

$$\begin{aligned}
 M(x, \alpha)^{\frac{1}{2}} - \min_{x \in \mathbb{R}^n} M(x, \alpha)^{\frac{1}{2}} &= M(x, \alpha)^{\frac{1}{2}} \\
 &\leq \sqrt{2\alpha(\alpha-1)} \|x - [x - F(x)]^+\| \\
 &= \sqrt{2\alpha(\alpha-1)} \|x - [x - F(x)]^+ - P_S[x] \\
 &\quad + [P_S[x] - F(P_S[x])]^+\| \\
 &\leq \sqrt{2\alpha(\alpha-1)} \|x - P_S[x]\| \\
 &\quad + \sqrt{2\alpha(\alpha-1)} \|[x - F(x)]^+ - [P_S[x] - F(P_S[x])]^+\| \\
 &\leq \sqrt{2\alpha(\alpha-1)} (2\|x - P_S[x]\| + \|F(x) - F(P_S[x])\|) \\
 &\leq \sqrt{2\alpha(\alpha-1)} (2 + \mu) d(x, S),
 \end{aligned}$$

where the first inequality follows from (2.6), the equality follows from $P_S[x] \in S$, the second inequality follows from the Cauchy–Schwarz inequality, the third from the nonexpansiveness of $P_S[\cdot]$, and the last from the Lipschitz continuity of $F(\cdot)$.

It is further known that, when $F(\cdot)$ is affine [12, 23] or $F(\cdot)$ has certain strong monotonicity structure [27, Theorem 2], then the following error bound holds:

$$d(x, S) \leq \rho \|x - [x - F(x)]^+\| \quad \forall x \text{ with } \|x - [x - F(x)]^+\| \leq \epsilon,$$

where ρ and ϵ are positive constants (independent of x). In that case, from (2.6),

$$(2.7) \quad M(x, \alpha) - \min_{x \in \mathbb{R}^n} M(x, \alpha) = M(x, \alpha) \geq \nu d(x, S)^2, \quad \nu := 2(\alpha - 1)/\rho^2 > 0.$$

By a well-known lemma [22, p. 6], we have that

$$M(x, \alpha) - M(P_S[x], \alpha) \leq \langle \nabla M(x, \alpha), x - P_S[x] \rangle + \frac{L}{2} \|x - P_S[x]\|^2.$$

Hence, from (2.7),

$$\begin{aligned}
 \langle \nabla M(x, \alpha), x - P_S[x] \rangle &\geq M(x, \alpha) - \frac{L}{2} d(x, S)^2 \\
 &\geq (\nu - L/2) M(x, \alpha).
 \end{aligned}$$

Therefore, if $\nu > L/2$,

$$\begin{aligned}
 \|\nabla M(x, \alpha)\| d(x, S) &\geq \langle \nabla M(x, \alpha), x - P_S[x] \rangle \\
 &\geq (\nu - L/2) M(x, \alpha) \\
 &\geq \nu(\nu - L/2) d(x, S)^2.
 \end{aligned}$$

Hence

$$\|\nabla M(x, \alpha)\| \geq \nu(\nu - L/2) d(x, S).$$

Since in this case $C = \mathbb{R}^n$, it follows that $r(x) = \nabla M(x, \alpha)$ and (2.4) holds.

3. Concluding remarks. It was established that the parallel variable distribution approach can be successfully applied to solve optimization problems with general convex constraints. Furthermore, conditions imposed on solving parallel subproblems were considerably relaxed, thus yielding a more practical framework. A rate of convergence result was also presented.

Acknowledgment. I am grateful to the two anonymous referees for careful reading and constructive suggestions which led to numerous improvements in the paper.

REFERENCES

- [1] D. P. BERTSEKAS, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.
- [2] D. P. BERTSEKAS AND J. N. TSITSIKLIS, *Parallel and Distributed Computation*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1989.
- [3] R. W. COTTLE, F. GIANNESSI, AND J.-L. LIONS, *Variational Inequalities and Complementarity Problems: Theory and Applications*, Wiley, New York, 1980.
- [4] M. C. FERRIS AND O. L. MANGASARIAN, *Parallel variable distribution*, SIAM J. Optim., 4 (1994), pp. 815–832.
- [5] A. V. FIACCO AND G. P. MCCORMICK, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley & Sons, New York, 1968.
- [6] R. FLETCHER, *Practical Methods of Optimization*, John Wiley & Sons, Chichester, UK, 1981.
- [7] M. FUKUSHIMA, *Parallel variable transformation in unconstrained optimization*, SIAM J. Optim., 8 (1998), to appear.
- [8] S.-P. HAN AND O. L. MANGASARIAN, *A dual differentiable exact penalty function*, Math. Programming, 25 (1983), pp. 293–301.
- [9] Y. Y. LIN AND J.-S. PANG, *Iterative methods for large convex quadratic programs: A survey*, SIAM J. Control Optim., 25 (1987), pp. 383–411.
- [10] X.-D. LUO AND P. TSENG, *On global projection-type error bound for the linear complementarity problem*, Linear Algebra Appl., to appear.
- [11] Z.-Q. LUO, O. L. MANGASARIAN, J. REN, AND M. V. SOLODOV, *New error bounds for the linear complementarity problem*, Math. Oper. Res., 19 (1994), pp. 880–892.
- [12] Z.-Q. LUO AND P. TSENG, *Error bound and convergence analysis of matrix splitting algorithms for the affine variational inequality problem*, SIAM J. Optim., 2 (1992), pp. 43–54.
- [13] Z.-Q. LUO AND P. TSENG, *On the linear convergence of descent methods for convex essentially smooth minimization*, SIAM J. Control Optim., 30 (1992), pp. 408–425.
- [14] Z.-Q. LUO AND P. TSENG, *Error bounds and convergence analysis of feasible descent methods: A general approach*, Ann. Oper. Res., 46 (1993), pp. 157–178.
- [15] O. L. MANGASARIAN, *Nonlinear Programming*, McGraw-Hill, New York, 1969.
- [16] O. L. MANGASARIAN, *Parallel gradient distribution in unconstrained optimization*, SIAM J. Control Optim., 33 (1995), pp. 1916–1925.
- [17] O. L. MANGASARIAN AND J. REN, *New improved error bounds for the linear complementarity problem*, Math. Programming, 66 (1994), pp. 241–255.
- [18] O. L. MANGASARIAN AND M. V. SOLODOV, *Nonlinear complementarity as unconstrained and constrained minimization*, Math. Programming, 62 (1993), pp. 277–297.
- [19] J. M. ORTEGA AND W. C. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.
- [20] J.-S. PANG, *A posteriori error bounds for the linearly-constrained variational inequality problem*, Math. Oper. Res., 12 (1987), pp. 474–484.
- [21] J.-S. PANG, *Complementarity problems*, in Handbook of Global Optimization, R. Horst and P. Pardalos, eds., Kluwer Academic Publishers, Norwell, MA, 1995, pp. 271–338.
- [22] B. T. POLYAK, *Introduction to Optimization*, Optimization Software, Inc., Publications Division, New York, 1987.
- [23] S. M. ROBINSON, *Some continuity properties of polyhedral multifunctions*, Math. Programming Study, 14 (1981), pp. 206–214.
- [24] R. T. ROCKAFELLAR, *Augmented Lagrange multiplier functions and duality in nonconvex programming*, SIAM J. Control, 12 (1974), pp. 268–285.
- [25] M. V. SOLODOV, *New inexact parallel variable distribution algorithms*, Comput. Optim. Appl., 7 (1997), pp. 165–182.
- [26] P. TSENG, *Dual coordinate ascent methods for non-strictly convex minimization*, Math. Programming, 59 (1993), pp. 231–248.
- [27] P. TSENG, *On linear convergence of iterative methods for the variational inequality problem*, J. Comput. Appl. Math., 60 (1995), pp. 237–252.

A GENERALIZED PROXIMAL POINT ALGORITHM FOR THE VARIATIONAL INEQUALITY PROBLEM IN A HILBERT SPACE*

REGINA S. BURACHIK[†] AND ALFREDO N. IUSEM[‡]

Abstract. We consider a generalized proximal point method for solving variational inequality problems with monotone operators in a Hilbert space. It differs from the classical proximal point method (as discussed by Rockafellar for the problem of finding zeroes of monotone operators) in the use of generalized distances, called Bregman distances, instead of the Euclidean one. These distances play not only a regularization role but also a penalization one, forcing the sequence generated by the method to remain in the interior of the feasible set so that the method becomes an interior point one. Under appropriate assumptions on the Bregman distance and the monotone operator we prove that the sequence converges (weakly) if and only if the problem has solutions, in which case the weak limit is a solution. If the problem does not have solutions, then the sequence is unbounded. We extend similar previous results for the proximal point method with Bregman distances which dealt only with the finite dimensional case and which applied only to convex optimization problems or to finding zeroes of monotone operators, which are particular cases of variational inequality problems.

Key words. convex optimization, variational inequalities, proximal point methods, monotone operators

AMS subject classifications. 90C25, 90C30

PII. S1052623495286302

1. Introduction. In this paper we are concerned with proximal algorithms for solving variational inequality problems. Let H be a Hilbert space. The operator $T : H \rightarrow \mathcal{P}(H)$ is *monotone* if

$$\langle u - v, x - y \rangle \geq 0$$

for all $x, y \in H$ and all $u \in T(x), v \in T(y)$. A monotone operator is called *maximal* if for any other monotone operator $\tilde{T}$ with $\tilde{T}(x) \supseteq T(x)$ for all $x \in H$, then $\tilde{T} = T$. Given a closed and convex set $C \subset H$, the variational inequality problem $VIP(T, C)$ consists of finding $\bar{x} \in C$ such that there exists $\bar{u} \in T(\bar{x})$ satisfying

$$(1) \quad \langle \bar{u}, x - \bar{x} \rangle \geq 0$$

for all $x \in C$. We will denote by S^* the set of solutions of $VIP(T, C)$.

An in-depth study of variational inequalities can be found in [16], and a survey of the subject and its connections to optimization theory can be found in [12].

Observe that when $C = H$, $VIP(T, C)$ reduces to the problem of finding a zero of T , i.e., an $x \in H$ such that $0 \in T(x)$.

The proximal point algorithm (PPA) for the problem of finding zeroes of an operator T in a Hilbert space generates a sequence $\{x^k\} \subset H$ in the following way: It starts with any $x^0 \in H$ and, given x^k , x^{k+1} is taken so that $0 \in T_k(x^{k+1})$ where $T_k(x) = T(x) + \lambda_k(x - x^k)$ and $\lambda_k \in (0, \tilde{\lambda})$ for some $\tilde{\lambda} > 0$. See [18] for a modern

*Received by the editors May 18, 1995; accepted for publication (in revised form) November 4, 1996.

<http://www.siam.org/journals/siopt/8-1/28630.html>

[†]Engenharia de Sistemas e Computação, COPPE/UFRJ, CP 68511 Rio de Janeiro, RJ, 21945-970, Brazil (regi@cos.ufrj.br). The research of this author was supported by CNPq grant 301280/94-0.

[‡]Instituto de Matemática Pura e Aplicada, Estrada Dona Castorina, 110, Rio de Janeiro, RJ, CEP 22460-320, Brazil (iusp@impa.br). The research of this author was partially supported by CNPq grant 301280/86.

survey on the method. The PPA can be seen as a regularization method in which the regularization parameter need not approach 0, thus avoiding a possible ill behavior of the regularized problems.

The main convergence results for PPA can be found in [22] where it is proved that $\{x^k\}$ converges weakly to a zero of T , provided that the set of zeroes is nonempty. It is well known that $VIP(T, C)$ is equivalent to finding a zero of the operator $\tilde{T} = T + N_C$, where N_C is the normality operator associated to the set C (see definition in section 5). Therefore, PPA can be used to solve $VIP(T, C)$. In the k th iteration, x^{k+1} must be a zero of $T_k + N_C$ with T_k as above. Equivalently, x^{k+1} must be the solution of $VIP(T_k, C)$. So, PPA solves the variational inequality problem by solving a sequence of variational inequality problems (with the operators T_k).

Although the operators T_k are better conditioned in principle than T (for example $VIP(T_k, C)$ has a unique solution when $VIP(T, C)$ has solutions), the subproblems are structurally as hard to solve as the original problem. In this paper, we consider a generalized proximal point algorithm (GPPA) for $VIP(T, C)$ (where C has nonempty interior) which generates subproblems which are structurally simpler than the original problem, as we show below.

In GPPA, we start with a strictly convex $g : C \rightarrow \mathbf{R}$ whose gradient diverges at the boundary ∂C of C (some additional technical conditions on the behavior of g near ∂C are required; see section 2). Now we define $T_k(x) = T(x) + \lambda_k(\nabla g(x) - \nabla g(x^k))$. Strict convexity of g and divergence of ∇g at ∂C guarantee uniqueness and existence of a zero of T_k , which belongs to the interior C° of C . Such a zero is automatically a solution of $VIP(T_k, C)$, but in our method we can just take x^{k+1} as the zero of T_k without any explicit consideration of the constraint set C , which has been embedded into g (which is in fact an interior penalty function tailored for the set C). In fact, we end up with genuinely unconstrained subproblems.

As an example, we can consider the case in which T is point-to-point, $H = \mathbf{R}^n$, and C is the nonnegative orthant $\mathbf{R}_+^n$. The function $g(x) = \sum_{j=1}^n \log x_j$, continuously extended to $\partial \mathbf{R}_+^n$ with the convention $0 \log 0 = 0$, satisfies all the required assumptions. Then the regularized operator of our algorithm at iteration k is $T_k(x) = T(x) + \lambda_k(\nabla g(x) - \nabla g(x^k))$ so that $T(x)_j = T(x)_j + \lambda_k(\log x_j - \log x_j^k)$.

It is well known that for $C = \mathbf{R}_+^n$, $VIP(T, C)$ reduces to the nonlinear complementarity problem (NCP): Find $x \in \mathbf{R}^n$ such that

$$(2) \quad x^t T(x) = 0,$$

$$(3) \quad x \geq 0, \quad T(x) \geq 0.$$

The PPA applied to this problem generates a subproblem of the form

$$(4) \quad x^t (T(x) + \lambda_k(x - x^k)) = 0,$$

$$(5) \quad x \geq 0, \quad T(x) + \lambda_k x \geq \lambda_k x^k,$$

while GPPA, in view of the formula for T_k given above, reduces to the following system of nonlinear equations:

$$(6) \quad T(x)_j + \lambda_k \log x_j = \lambda_k \log x_j^k, \quad j = 1, \dots, n.$$

The difference between PPA and GPPA in this example is clear. The PPA subproblems are NCPs of the same nature as the original problem, which is considerably

harder to solve, from a computational point of view, than system (6). For instance, if T is continuously differentiable, (6) can be easily solved with Newton's method, while (4)–(5) has the additional combinatorial complication of determining the sets of zero components of x .

Another way of looking at PPA and GPPA is the following: For PPA, the factor $x - x^k$ in the definition of T_k is the gradient of $\rho_k(x) = \|x - x^k\|^2$ so that, when $T(x) = \partial f(x)$ for some convex f , $T_k = \partial f_k(x)$ with $f_k(x) = f(x) + \lambda_k \|x - x^k\|^2$ and $VIP(T, C)$ reduces to $\min f_k(x)$ subject to $x \in C$. In the case of GPPA, we can consider $D_g : C \times C^o \rightarrow \mathbf{R}$, defined as $D_g(x, y) = g(x) - g(y) - \langle \nabla g(y), x - y \rangle$. If we define now $\rho_k(x) = D_g(x, x^k)$, we get, from the definition of D_g , $\partial \rho_k(x) = \nabla g(x) - \nabla g(x^k)$ so that we again have $T_k(x) = T(x) + \lambda_k \partial \rho_k(x)$, and in the optimization case (i.e., $T(x) = \partial f(x)$) a zero of T_k is just a minimizer of $f_k(x) = f(x) + \lambda_k D_g(x, x^k)$. Such a minimizer is unique and belongs to C^o due to the properties of g . $D_g(\cdot, \cdot)$ shares some of the properties of $\|\cdot - \cdot\|^2$ mentioned in section 2. In this situation, g is said to be a *Bregman function* and D_g the *Bregman distance* associated with g . The effective domain of ∂g is said to be the *zone* of g (in our case the zone of g is C^o).

Bregman distances, which originate in [1], have been considered, in connection with proximal point methods for convex optimization, by several authors. In [10], the method is studied for g as in the above-mentioned example. Later, convergence results for the method with a general Bregman distance for convex optimization were obtained first in [6], then under weaker assumptions in [8] and under even weaker ones in [13]. In this paper we will use the assumptions on D_g introduced in [13]. Very recently, a more general approach was developed in [17] which removes two further assumptions on the function g (continuity in the boundary of S and differentiability in the interior of S). This work, as those mentioned above, deals only with the convex optimization problem and not with variational inequalities. We conjecture that our results for VIP can also be obtained with functions g as those considered in [17].

The proximal point methods with Bregman distances for finding zeroes of monotone operators has been considered in [9] under the restrictive assumptions on g used in [6]. All these works deal only with the finite-dimensional case.

We emphasize that the ability to deal with $VIP(T, C)$ rather than finding zeroes of T is essential for getting subproblems which are structurally easier than the original problem. None of the works mentioned above cover the case of (6), i.e., the use of GPPA for (2)–(3). In addition, our results hold in a general Hilbert space.

Our main convergence result is the following: Under appropriate assumptions GPPA generates a sequence that converges to $x^* \in S^*$ if and only if $S^* \neq \emptyset$.

For the infinite-dimensional case and an arbitrary Bregman distance, we obtain first a “partial convergence” result, meaning that the sequence is bounded, the distance between consecutive iterates goes to 0, and every weak cluster point of the sequence is a solution of the problem, and then a full weak convergence result (weak convergence of the whole sequence to a solution of the problem) under a stronger hypothesis on the Bregman distance, namely, norm compatibility (see condition B_6 below). This condition holds automatically in the finite-dimensional case.

2. Bregman distances. Let $S \subset H$ be an open and convex set. We denote by $\bar{S}$ the closure of S . Take $g : H \rightarrow \mathbf{R}$ Gâteaux differentiable and define $D_g : \bar{S} \times S \rightarrow \mathbf{R}$ as

$$(7) \quad D_g(x, y) = g(x) - g(y) - \langle \nabla g(y), x - y \rangle$$

where $\nabla g(y)$ is the Gâteaux derivative of g at y .

DEFINITION 1. The function g is said to be a Bregman function with zone S and D_g the Bregman distance induced by g if the following conditions hold:

- (B₁) g is weakly lower semicontinuous on $\bar{S}$.
- (B₂) g is strictly convex, continuous in $\bar{S}$, and continuously differentiable in S with respect to the strong topology.
- (B₃) The sets $\Gamma_1(y, \delta) = \{x \in \bar{S} | D_g(x, y) \leq \delta\}$ and $\Gamma_2(x, \delta) = \{y \in S | D_g(x, y) \leq \delta\}$ are bounded for all $\delta > 0$, $y \in S$, and $x \in \bar{S}$, respectively.
- (B₄) If $\{x^k\}, \{y^k\} \subset S$, $x^k \xrightarrow{w} x$, $y^k \xrightarrow{w} x$, and $\lim_{k \rightarrow \infty} D_g(x^k, y^k) = 0$, then $\lim_{k \rightarrow \infty} [D_g(x, x^k) - D_g(x, y^k)] = 0$.
- (B₅) If $\{x^k\} \subset \bar{S}$ is bounded, $\{y^k\} \subset S$ is such that $y^k \xrightarrow{w} y$, and $\lim_{k \rightarrow \infty} D_g(x^k, y^k) = 0$, then $x^k \xrightarrow{w} y$.

Observe that $D_g(x, y) \geq 0$ for any $x, y \in H$ and $D_g(x, y) = 0$ if and only if $x = y$ by B₂. We introduce next three subclasses of Bregman functions which will be used in our analysis. The first one holds automatically in the finite-dimensional case, as we will show later on. A Bregman function g is said to be *norm compatible* if, in addition to B₁–B₅, it satisfies the following:

- (B₆) If $\{x^k\} \subset S$ and $\{y^k\} \subset S$ are such that $x^k \xrightarrow{w} x$ and $y^k \xrightarrow{w} y$, with $x \neq y$, then $\liminf_{k \rightarrow \infty} |\langle \nabla g(x^k) - \nabla g(y^k), x - y \rangle| > 0$.

The following two conditions are needed in order to keep the sequence generated by the method in S .

The Bregman function g is said to be *zone coercive* if, in addition to B₁–B₅, the following condition holds:

- (B₇) For every $y \in H$, there exists an $x \in S$ such that $\nabla g(x) = y$.

The Bregman function g is said to be *boundary coercive* if in addition to B₁–B₅ the following condition holds:

- (B₈) If $\{y^k\} \subset S$ is weakly convergent to some point in the boundary of S and x belongs to S , then $\lim_{k \rightarrow \infty} D_g(x, y^k) = +\infty$.

Boundary coerciveness is closely related to *essential smoothness* as defined in [20]. A proof of the fact that B₇ implies B₈ can be found in Corollary 1 of [7]. We present next the main property of boundary coercive Bregman functions.

In order to study the subdifferential properties of $D_g(\cdot, y)$ (with $y \in S$) we will assume from now on that both g and $D_g(\cdot, y)$ have been extended in the usual way, i.e., taking $g(x) = D_g(x, y) = +\infty$ for all $x \notin \bar{S}$. For a convex $f : H \rightarrow \mathbf{R}$, the point-to-set operator ∂f will denote the subdifferential of f (i.e., $\partial f(x)$ is the set of subgradients of f at x) and $\partial_x D_g(\cdot, y)$ will denote the subdifferential of D_g as a function of its first argument.

LEMMA 1. If g is a boundary coercive or a zone coercive function with zone S , then, for any fixed $y \in S$,

$$\partial_x D_g(x, y) = \begin{cases} \nabla g(x) - \nabla g(y) & \text{if } x \in S, \\ \emptyset & \text{otherwise.} \end{cases}$$

Proof. The value of $\partial_x D_g(x, y)$ for $x \in S$ follows easily from (7). It is also clear that $\partial_x D_g(x, y) = \emptyset$ for $x \notin \bar{S}$. Therefore, we only need to consider the case of x belonging to the boundary ∂S of S . For the sake of simplicity of notation, let $f_y(\cdot) = D_g(\cdot, y)$. Take $x \in \partial S$ and assume that there exists $\xi \in \partial f_y(x)$. Note that f_y differs from g just in a linear term so that f_y is continuous and strictly convex in $\bar{S}$ by B₁ and B₂.

If g is zone coercive, then by B₇ there exists $z \in S$ such that $\nabla g(z) = \xi + \nabla g(y)$ so that $\nabla f_y(z) = \nabla g(z) - \nabla g(y) = \xi$, implying $0 = \langle \xi - \xi, z - x \rangle = \langle \nabla f_y(z) - \xi, z - x \rangle$

with $\xi \in \partial f_y(x)$. By strict convexity of f_y we get $z = x$, which is a contradiction because $z \in S$, $x \in \bar{S}$, and S is open. It follows that $\partial f_y(x) = \emptyset$.

Now we consider the case of a boundary coercive g . Take a sequence $\{\varepsilon_k\} \subset (0, 1)$ such that $\lim_{k \rightarrow \infty} \varepsilon_k = 0$ and let $x^k = (1 - \varepsilon_k)x + \varepsilon_k y$. By convexity of S , $x^k \in S$ for all k and, using the definition of subgradient,

$$\begin{aligned} \varepsilon_k \langle \xi, y - x \rangle &= \langle \xi, x^k - x \rangle \\ &\leq f_y(x^k) - f_y(x) \leq \langle \nabla f_y(x^k), x^k - x \rangle \\ (8) \quad &= \langle \nabla g(x^k) - \nabla g(y), x^k - x \rangle. \end{aligned}$$

By definition of x^k , $x^k - x = \varepsilon_k/(1 - \varepsilon_k)(y - x^k)$. It follows from (7) that

$$\begin{aligned} &\langle \nabla g(u) - \nabla g(v), u - v \rangle \\ &= D_g(u, v) + D_g(v, u) \end{aligned}$$

for all $u, v \in S$. Then using these two facts in (8),

$$\begin{aligned} \varepsilon_k \langle \xi, y - x \rangle &\leq \frac{\varepsilon_k}{1 - \varepsilon_k} \langle \nabla g(x^k) - \nabla g(y), y - x^k \rangle \\ (9) \quad &= \frac{-\varepsilon_k}{1 - \varepsilon_k} (D_g(x^k, y) + D_g(y, x^k)). \end{aligned}$$

From (9)

$$(10) \quad (1 - \varepsilon_k) \langle \xi, y - x \rangle + f_y(x^k) \leq -D_g(y, x^k).$$

Since $x^k \xrightarrow{w} x \in \partial S \subset \bar{S}$ and f_y is weakly lower semicontinuous in $\bar{S}$, the limit of the left-hand side of (10) as k goes to ∞ is greater than or equal to $\langle \xi, y - x \rangle + D_g(x, y)$. By B_8 the limit of the right-hand side of (10) is $-\infty$. This contradiction establishes that $\partial f_y(x) = \emptyset$. $\square$

We give next some examples of Bregman functions. $\mathbf{R}_{++}^n$ will denote the interior of the nonnegative orthant of $\mathbf{R}^n$.

Example 1. $S = H$, $g(x) = \|x\|^2$. In this case $D_g(x, y) = \|x - y\|^2$. The fact that this g satisfies B_1 – B_8 in an arbitrary Hilbert space has been proved in [5].

Example 2. $S = \mathbf{R}_{++}^n$, $g(x) = \sum_{j=1}^n x_j \log x_j$, extended with continuity to the boundary of $\mathbf{R}_{++}^n$ with the convention that $0 \log 0 = 0$. In this case

$$D_g(x, y) = \sum_{j=1}^n \left(x_j \log \frac{x_j}{y_j} + y_j - x_j \right),$$

which is the Kullback–Leibler divergence, which is widely used in statistics.

Example 3. $S = \mathbf{R}_{++}^n$, $g(x) = \sum_{j=1}^n (x_j^\alpha - x_j^\beta)$ with $\alpha \geq 1$, $0 < \beta < 1$. For $\alpha = 2$, $\beta = \frac{1}{2}$ we get $D_g(x, y) = \|x - y\|^2 + \sum_{j=1}^n \frac{1}{2\sqrt{y_j}} (\sqrt{x_j} - \sqrt{y_j})^2$, and for $\alpha = 1$, $\beta = \frac{1}{2}$ we get $D_g(x, y) = \sum_{j=1}^n \frac{1}{2\sqrt{y_j}} (\sqrt{x_j} - \sqrt{y_j})^2$.

The Bregman functions of Examples 1, 2, and 3 are all boundary coercive. They are also zone coercive, excepting the case of Example 3 with $\alpha = 1$. Verification of the properties of Bregman functions for Examples 2 and 3 is a matter of routine. Other examples of boundary and zone coercive Bregman functions of interest for applications can be found in [7] and [13], which deal only with the finite-dimensional case.

3. Paramonotonicity, pseudomonotonicity, and regularity. Our convergence theorems require two conditions on the operator T , namely para and pseudomonotonicity, which we discuss next. The notion of paramonotonicity was introduced in [7] and further studied in [14]. It is defined as follows.

DEFINITION 2. T is *paramonotone* in a convex set C if it is monotone and $\langle z - z', w - w' \rangle = 0$ with $z, z' \in C$, $w \in T(z)$, $w' \in T(z')$, implies $w \in T(z')$, $w' \in T(z)$.

The next proposition presents the main properties of paramonotone operators.

PROPOSITION 1.

(i) If T is the subdifferential ∂f of a convex function $f : H \rightarrow \mathbf{R}$, then T is paramonotone in H .

(ii) If T is paramonotone in C , x^* solves $VIP(T, C)$, and $\bar{x} \in C$ satisfies that there exists an element $\bar{u} \in T(\bar{x})$ such that $\langle \bar{u}, x^* - \bar{x} \rangle \geq 0$, then $\bar{x}$ also solves $VIP(T, C)$.

(iii) If T_1 and T_2 are paramonotone in C then $T_1 + T_2$ is paramonotone in C .

Proof.

(i) Monotonicity of T is well known. Assume that $\langle u - v, x - y \rangle = 0$ with $x \in C$, $y \in C$, $u \in T(x)$, $v \in T(y)$, and define $\bar{f} : C \rightarrow \mathbf{R}$ as $\bar{f}(z) = f(z) + \langle u, x - z \rangle$. Then $\bar{f}$ is convex and $\partial \bar{f}(z) = \partial f(z) - u = \{w - u : w \in \partial f(z)\}$. Taking $w = u$, we get that $0 \in \partial \bar{f}(x)$ and so x is an unrestricted minimizer of $\bar{f}$. By hypothesis and definition of subgradients, $f(x) - f(y) \leq \langle u, x - y \rangle = \langle v, x - y \rangle \leq f(x) - f(y)$, implying $f(x) = f(y) + \langle u, x - y \rangle = \bar{f}(y)$. Since $\bar{f}(x) = f(x)$ by definition of $\bar{f}$, we conclude that $\bar{f}(x) = \bar{f}(y)$ so that y is also an unrestricted minimizer of $\bar{f}$; i.e., $0 \in \partial \bar{f}(y)$. Therefore, $0 = w - u$ for some $w \in \partial f(y)$, which is equivalent to $u \in \partial f(y) = T(y)$. Reversing the roles of (x, u) , (y, v) the same argument proves that $v \in T(x)$ and the result is established in view of Definition 2.

(ii) Assume that $\langle \bar{u}, x^* - \bar{x} \rangle \geq 0$ for some $\bar{u} \in T(\bar{x})$ and some solution x^* of $VIP(T, C)$. Since x^* is a solution of $VIP(T, C)$, there exists $u^* \in T(x^*)$ such that

$$(11) \quad \langle u^*, x - x^* \rangle \geq 0$$

for all $x \in C$. Then by monotonicity of T , $0 \leq \langle \bar{u}, x^* - \bar{x} \rangle \leq \langle u^*, x^* - \bar{x} \rangle \leq 0$, implying that

$$(12) \quad \langle \bar{u}, x^* - \bar{x} \rangle = \langle u^*, x^* - \bar{x} \rangle = 0.$$

From the last equality, $\langle \bar{u} - u^*, x^* - \bar{x} \rangle = 0$, and by paramonotonicity of T we get

$$(13) \quad u^* \in T(\bar{x}).$$

Then, for all $x \in C$

$$\langle u^*, x - \bar{x} \rangle = \langle u^*, x - x^* \rangle + \langle u^*, x^* - \bar{x} \rangle = \langle u^*, x - x^* \rangle \geq 0$$

using (12) in the second equation and (11) in the inequality. It follows from (13) and the rightmost inequality of the last expression that $\bar{x}$ is a solution of $VIP(T, C)$.

(iii) This property follows easily from Definition 2. $\square$

Other conditions guaranteeing paramonotonicity, besides Proposition 1(i), can be found in [13].

For a point-to-set operator T defined in H , $D(T)$ will denote its domain; i.e., $D(T) = \{x \in H | T(x) \neq \emptyset\}$.

Next we recall the definition of pseudomonotonicity, which has been taken from [4] and should not be confused with other uses of the same word, e.g., [15].

DEFINITION 3. Let H be a Hilbert space and $T : H \rightarrow \mathcal{P}(H)$ such that $D(T)$ is closed and convex. T is said to be pseudomonotone if and only if it satisfies the following condition.

Take any sequence $\{x^k\} \subset D(T)$, converging weakly to an element $x^0 \in D(T)$, and any sequence $\{w^k\} \subset H$, with $w^k \in T(x^k)$ for all k , such that

$$\limsup_k \langle w^k, x^k - x^0 \rangle \leq 0.$$

Then for each $y \in D(T)$ there exists an element $w^0 \in T(x^0)$ such that

$$\langle w^0, x^0 - y \rangle \leq \liminf_k \langle w^k, x^k - y \rangle.$$

The next proposition lists several conditions which ensure pseudomonotonicity. We recall now the classical definition of upper semicontinuity.

DEFINITION 4. Let S and S_1 be two topological spaces and T a mapping of S into $\mathcal{P}(S_1)$. Then T is said to be an upper semicontinuous set valued mapping of S if for each point $s_0 \in S$ and each open neighborhood V of $T(s_0)$ in S_1 there exists a neighborhood U of s_0 in S (with U depending on V) such that $T(U) \subset V$.

PROPOSITION 2. Consider an operator $T : H \rightarrow \mathcal{P}(H)$. If any of the following conditions hold, then T is pseudomonotone from $D(T)$ to $\mathcal{P}(H)$.

(i) $D(T)$ is closed and convex, $T(x)$ is closed and convex for all $x \in D(T)$, and T is upper semicontinuous as a set valued function from each line segment in $D(T)$ to H with respect to the weak topology,

(ii) $H = \mathbf{R}^n$ and T is point-to-point and hemicontinuous; i.e., for all $x, y \in \mathbf{R}^n$ the map $\varphi(t)$ defined as $\varphi(t) = \langle T((1-t)x + ty), x - y \rangle$ is continuous,

(iii) $T = \nabla f$ with $f : H \rightarrow \mathbf{R}$ convex and Gâteaux differentiable.

Proof.

(i) See [4].

(ii) See [19, p. 107].

(iii) Follows from (ii) and the fact that the gradient of a convex and Gâteaux differentiable function is hemicontinuous, proved in [19, p. 94]. $\square$

It is easy to verify that every point-to-point and continuous operator is pseudomonotone. In particular, if we take $T(x) = Ax$ with $A \in \mathbf{R}^{n \times n}$ such that $(A + A^t)$ is positive semidefinite and $A \neq A^t$, then we get an example of a monotone and pseudomonotone operator which is not the subdifferential of any convex function.

We need now the notion of regularity, a property of maximal monotone operators introduced in [2]. Let $G(T)$ denote the graph of T ; i.e., $G(T) = \{(y, v) | y \in H, v \in T(y)\}$.

DEFINITION 5. A maximal monotone operator $T : H \rightarrow \mathcal{P}(H)$ is said to be regular if and only if for all $u \in R(T)$ and for all $x \in D(T)$ it holds that $\sup_{(y,v) \in G(T)} \langle v - u, x - y \rangle < \infty$.

For a maximal monotone operator T , let $R(T)$ denote the range of T ; i.e., $R(T) = \bigcup_{x \in H} T(x)$.

For any set V , $\text{conv}(V)$ will denote the convex hull of V . The following results have been proved in [2, Lemma 1].

LEMMA 2. Let $T_0 : H \rightarrow \mathcal{P}(H)$ be a maximal monotone operator and $F \subset H$. If for all $u \in F$ there exists $x \in H$ such that

$$(14) \quad \sup_{(v,y) \in G(T_0)} \langle v - u, x - y \rangle < \infty,$$

then

- (i) $\text{conv}(F) \subseteq \overline{R(T_0)}$,
- (ii) $(\text{conv}(F))^0 \subseteq R(T_0)$.

PROPOSITION 3. *Let T_1, T_2 be monotone operators. Suppose that they satisfy the following conditions:*

- (a) T_1 is regular,
- (b) $D(T_1) \cap D(T_2) \neq \emptyset$ and $R(T_1) = H$,
- (c) $T_1 + T_2$ is maximal monotone.

Then $R(T_1 + T_2) = H$.

Proof. We apply Lemma 2 to $F := R(T_1) + R(T_2)$ and $T_0 := T_1 + T_2$. The result follows from Lemma 2(ii) and assumption (b) if we prove that F satisfies (14). We proceed to verify (14) for $F = R(T_1) + R(T_2)$. Let $u \in R(T_2) + R(T_1)$, and take $x \in D(T_1) \cap D(T_2)$ and $w \in T_2(x)$. Then $u = w + (u - w)$ for $w \in T_2(x)$.

Since $R(T_1) = H$, there exists $y \in H$ such that $u - w \in T_1(y)$. Then, since T_1 is regular, given $u - w \in R(T_1)$ and $x \in D(T_1)$, there exists some $\gamma \in \mathbf{R}$ such that

$$\sup_{(z,s) \in G(T_1)} \langle s - (u - w), x - z \rangle \leq \gamma.$$

Then, for any $(z, s) \in G(T_1)$, we have

$$(15) \quad \langle s - (u - w), x - z \rangle \leq \gamma.$$

Take $v \in T_2(z)$. For any $z \in D(T_2) \cap D(T_1)$, by monotonicity of T_2 , we have

$$(16) \quad \langle v - w, z - x \rangle \geq 0$$

because $w \in T_2(x)$. Adding (15) and (16), we obtain

$$\langle (s + v) - u, x - z \rangle \leq \gamma$$

for any $s \in T_1(z)$, $v \in T_2(z)$, i.e., for any $s + v \in (T_1 + T_2)(z)$. Therefore,

$$\sup_{(z,t) \in G(T_1+T_2)} \langle t - u, x - z \rangle < +\infty$$

and (14) is established for $F = R(T_1) + R(T_2)$. $\square$

4. Statement of the algorithm. Let C be a closed and convex subset of a Hilbert space H , g a Bregman function with zone $S = C^0$ as defined in section 2, and $\{\lambda_k\}$ a sequence of positive real numbers bounded above by some $\tilde{\lambda} > 0$. We define algorithm GPPA as follows.

- (i) Initialization.

$$(17) \quad x^0 \in C^0.$$

(ii) Iterative step. Given x^k , define $T_k : H \rightarrow \mathcal{P}(H)$ as $T_k(\cdot) = T(\cdot) + \lambda_k \partial_x D_g(\cdot, x^k)$ and let $x^{k+1} \in H$ be such that

$$(18) \quad 0 \in T_k(x^{k+1}).$$

Note that existence of x^{k+1} satisfying (18) is not immediate at all and will be ensured only under some assumptions on T and g . This issue is the matter of the

next section. On the other hand, if such an x^{k+1} exists, it is unique as a consequence of B_2 (see the proof of Theorem 1).

In connection with our convergence analysis, we have attempted to cover as many relevant cases as possible, which produces a rather involved and convoluted proof, where we must draw from many technical results on monotone operators. We mention that the proof becomes much simpler if we relax our assumptions. For instance, assuming zone coerciveness of g instead of boundary coerciveness, existence of solutions for the subproblems becomes quite easy (see Theorem 1), but then we exclude several relevant examples of Bregman functions. In order to establish the result under weaker assumptions (Theorem 2), we must use finer results on the range and maximality of the sum of monotone operators. Also the proof is much easier when the operator is point-to-point, but then we exclude cases in which C has empty interior, e.g., linear programming. Actually, the algorithm we consider applies to cases in which $C = C_1 \cap C_2$ with $C_1^0 = \emptyset$, $C_2^0 \neq \emptyset$ by observing that $VIP(T, C) = VIP(\tilde{T}, C_2)$ with $\tilde{T} = T + N_{C_1}$. This algorithm can be used for $VIP(\tilde{T}, C_2)$, but $\tilde{T}$ is not point-to-point even when T is. With point-to-set operators we must require pseudomonotonicity of T with the ensuing technical complications. As a third example, paramonotonicity is not needed if we look just for zeroes of T in the interior of C (e.g., with $C = H$), which makes the proof much easier, as in [9]. But variational inequality problems are interesting, particularly when the solutions are in the boundary (e.g., linear programming, linear or nonlinear complementarity problems), and in any of these cases interior point methods become advantageous when compared with PPA, as discussed in the introduction. Finally, several technical complications (for instance, the norm compatibility assumption) arise from the infinite dimensionality of H .

5. Existence results. For proving that the sequence given by (17)–(18) is well defined and contained in C^0 , we will need some preliminary material. For any convex set X , $\text{ri}(X)$ will denote the relative interior of X . The following proposition is proved in [21, Theorem 1].

PROPOSITION 4. *Let $T_1, T_2 : H \rightarrow \mathcal{P}(H)$ be maximal monotone operators. Suppose that any of the following conditions hold:*

- (a) $D(T_1) \cap (D(T_2))^0 \neq \emptyset$,
- (b) $H = \mathbf{R}^n$ and $\text{ri}(D(T_1)) \cap \text{ri}(D(T_2)) \neq \emptyset$.

Then $T_1 + T_2$ is a maximal monotone operator.

Now we introduce a function that will be useful in the rest of the paper.

Let $T : H \rightarrow \mathcal{P}(H)$ be a maximal monotone operator and C a nonempty closed and convex subset of H such that $D(T) \cap C \neq \emptyset$. We define the function

$$h_{T,C} : D(T) \cap C \rightarrow \mathbf{R} \cup \{\infty\}$$

as

$$(19) \quad h_{T,C}(x) := \sup_{(v,y) \in G_C(T)} \langle v, x - y \rangle,$$

where $G_C(T) := \{(v, y) | v \in T(y), y \in C\}$.

This function has interesting properties, some of which are described in the following lemmas.

LEMMA 3. *Let T, C and $h_{T,C}$ be as given by the definition above. Then*

- (i) $h_{T,C}$ is convex in $D(T) \cap C$.
- (ii) $h_{T,C} \geq 0$ in $D(T) \cap C$.

Proof.

(i) $h_{T,C}(x)$ is the supremum of a family of affine transformations, which are, in particular, convex functions, and the supremum of a family of convex functions is always convex.

(ii) Take $y = x$ in (19). $\square$

Let S^* be the set of solutions of $VIP(T, C)$. For a closed and convex set $V \subset \mathbf{R}^n$, let δ_V be the indicator function of V ; i.e.,

$$\delta_V(x) = \begin{cases} 0 & \text{if } x \in V, \\ +\infty & \text{otherwise.} \end{cases}$$

We define the normality operator N_V of V as $N_V(x) = \partial\delta_V(x)$ (i.e., the set of subgradients of δ_V at x). It is easy to check that $N_V(x) = \emptyset$ if $x \notin V$, $N_V(x) = \{0\}$ if $x \in V^0$, and that $N_V(x)$ is a nonempty and closed positive cone for all $x \in V$ so that $D(N_V) = V$. Being the subdifferential of a closed convex function, N_V is maximal monotone. It is easy to see that S^* is precisely the set of zeroes of $T + N_C$.

LEMMA 4. *Consider $h_{T,C}$ as given by (19). Suppose that any of the following conditions hold:*

- (i) $H = \mathbf{R}^n$ and $\text{ri}(D(T)) \cap \text{ri}(C) \neq \emptyset$.
- (ii) H is an arbitrary Hilbert space and $D(T) \cap C^0 \neq \emptyset$.

Then $h_{T,C}$ has zeroes if and only if S^* is nonempty and in such a case S^* is precisely the set of zeroes of $h_{T,C}$.

Proof. We will see first that a zero of $h_{T,C}$ is a solution of $VIP(T, C)$. For simplicity, let us denote $h := h_{T,C}$. Let $A := T + N_C$ so that $D(A) = D(T) \cap C$. If $h(x) = 0$, then, by (19), we have that for any $y \in D(T) \cap C$, $v \in Ty$,

$$(20) \quad \langle v, x - y \rangle \leq 0.$$

On the other hand, if we take any $w \in N_C(y)$, then, for any $y \in C \cap D(T)$,

$$(21) \quad \langle w, x - y \rangle \leq 0$$

where we use the definition of N_C and the assumption that $x \in C$. Adding (20) and (21), we obtain

$$(22) \quad \langle 0 - (v + w), x - y \rangle \geq 0$$

for any $y \in D(T) \cap C = D(A)$ and any $v + w \in A(y) = T(y) + N_C(y)$. By Proposition 4, A is maximal monotone both in case (i) and in case (ii). Then this maximality property together with (22) yields

$$(23) \quad 0 \in A(x) = T(x) + N_C(x),$$

and (23) establishes that $x \in S^*$.

Reciprocally, suppose now that $x \in S^*$. Then for any $y \in C$, there exists $v \in Ty$ with

$$(24) \quad \langle v, y - x \rangle \geq 0.$$

Using (24) and monotonicity, we have that for any $w \in Ty$

$$(25) \quad \langle w, y - x \rangle \geq 0.$$

It follows from (25) that

$$(26) \quad h(x) = \sup_{(y,w) \in G_C(T)} \langle w, x - y \rangle \leq 0.$$

Inequality (26) and Lemma 3(ii) imply that $h(x) = 0$, as we wanted to prove. $\square$

Now we must establish some properties of the regularized operators T_k used in the algorithm.

LEMMA 5. *Take $y \in C^0$ and $\lambda > 0$, assume that g is a zone coercive Bregman function with zone C^0 (i.e., B_7 holds for g), and define $\hat{T}(\cdot) := T(\cdot) + \lambda \partial_x D_g(\cdot, y)$.*

- (i) $\hat{T}$ is maximal monotone,
- (ii) $R(\hat{T}) = H$.

Proof.

(i) The operator $\hat{T}$ is trivially monotone. Let $T_1 = T$ and $T_2 = \lambda \partial_x D_g(\cdot, y)$. By Lemma 1, $D(T_1) \cap (D(T_2))^0 = D(T) \cap C^0 \neq \emptyset$. By Proposition 4, $\hat{T} = T_1 + T_2$ is maximal monotone.

(ii) It is a consequence of Proposition 3, which can be applied because conditions (a), (b), and (c) hold for $T_2 = T$ and $T_1 = \lambda \partial_x D_g(\cdot, y)$. More precisely,

- (a) $\lambda \partial_x D_g(\cdot, y)$ is regular because subdifferential of closed, convex, and lower semicontinuous functions are regular (see [2]).
- (b) $D(T) \cap C^0 \neq \emptyset$ and $R(\partial_x D_g(\cdot, y)) = H$ by B_6 , and
- (c) $\hat{T}$ is maximal by (i).

Then $R(\hat{T}) = H$ by Proposition 3. $\square$

We need one more preliminary result to be used in the rest of the paper, which is proved in [3, Corollary 2.2].

PROPOSITION 5. *Let $T : H \rightarrow \mathcal{P}(H)$ be maximal monotone. If $D(T)$ is bounded, then T is onto.*

Now we present our existence results. Recall that an operator A is *strictly monotone* if and only if

$$\langle u - v, x - y \rangle > 0$$

for all $x, y \in H$, $x \neq y$ and $u \in A(x), v \in A(y)$.

THEOREM 1. *Let $\{x^k\}$ be the sequence given by (17)–(18). Suppose that the following conditions hold:*

- (i) $D(T) \cap C^0 \neq \emptyset$,
- (ii) g is zone coercive with zone C^0 (i.e., it satisfies B_7).

Then $\{x^k\}$ is well defined and contained in C^0 .

Proof. The proof will be performed by induction on k . $x^0 \in C^0$ by (17). Suppose that $x^k \in C^0$. Let $B_k(\cdot) := \lambda_k \partial_x D_g(\cdot, x^k)$. Then B_k is strictly monotone because g is strictly convex by B_2 . This implies strict monotonicity of $T_k := T + B_k$. By Lemma 5(ii), T_k has a zero in $D(T_k)$, which is unique by strict monotonicity. We call this zero x^{k+1} . We have to show that this zero belongs to C^0 . By Lemma 1, $D(T_k) = D(T) \cap C^0$, and since $x^{k+1} \in D(T_k)$ we obtain that $x^{k+1} \in C^0$. We conclude that $\{x^k\} \subset C^0$. $\square$

THEOREM 2. *Let $\{x^k\}$ be the sequence given by (17)–(18). Suppose that $D(T) \cap C^0 \neq \emptyset$. Suppose further that $h_{T,C}(x) < \infty$ for all $x \in C \cap D(T)$ and g is boundary coercive (i.e., it satisfies B_8). Then $\{x^k\}$ is well defined and contained in C^0 .*

Proof. Now we do not have the property $R(T_k) = H$, which is essential in Theorem 1 and is implied by zone coerciveness of g . This hypothesis is replaced here by

boundary coerciveness of g and the finiteness of $h_{T,C}$ in $C \cap D(T)$. Again, we perform the proof by induction: For $k = 0$, we have that $x^0 \in C^0$ by (17). By inductive hypothesis and (18), there exists $x^k \in D(T) \cap C^0$ with $0 \in T_{k-1}x^k$. So we have that, in particular, $x^k \in D(T) \cap C$, and by the assumption on $h_{T,C}$ it holds that $0 \leq h_{T,C}(x^k) < \infty$. For the sake of a simpler notation, let $h := h_{T,C}$. If $h(x^k) = 0$, then by Lemma 4(ii) $x^k \in S^*$. We claim that by taking $x^{k+1} := x^k$ we get $0 \in T_k(x^{k+1})$. In fact, using Lemma 1, $T_k(x^{k+1}) = T(x^k) + \lambda_k \partial_x D_g(x^k, x^k) = T(x^k)$, and it is easy to check that if $x \in D(T) \cap C^0 \cap S^*$, then x is a zero of T . So in this case $x^{k+1} = x^k$, which belongs to C^0 by inductive hypothesis.

Then we can suppose that $h(x^k) > 0$. In this case, we define the set

$$S_k := \left\{ x \in C \mid D_g(x, x^k) \leq \frac{h(x^k)}{\lambda_k} \right\}.$$

The hypotheses on g , namely B_1 , B_2 , and B_3 , imply that S_k is closed, convex, and bounded, respectively. Also, an element x will be in S_k^0 if and only if

$$(27) \quad x \in C^0 \text{ and } D_g(x, x^k) < \frac{h(x^k)}{\lambda_k}.$$

Observe that $x^k \in (S_k)^0$ because $D_g(x^k, x^k) = 0 < h(x^k)/\lambda_k$ and $x^k \in C^0$ by induction hypothesis. Let $N_k := N_{S_k}$, the normality operator of S_k . It is well known that $D(N_k) = S_k$, which is a bounded set by B_3 . Now we define $B_k(\cdot) := N_k(\cdot) + \lambda_k \partial_x D_g(\cdot, x^k)$. In order to prove that B_k is maximal monotone, we check condition (a) of Proposition 4. By the induction hypothesis and Lemma 1, $x^k \in D(N_k) \cap C^0 = S_k \cap D(\partial_x D_g(\cdot, x^k))$. Then $D(N_k) \cap (D(\nabla_x D_g(\cdot, x^k)))^0 \neq \emptyset$ and so (a) of Proposition 4 holds. So B_k is maximal. Also, $D(B_k)$ is bounded, being a subset of S_k , and using Proposition 5 we conclude that $R(B_k) = H$. Let us consider now the operator $A_k =: T + B_k$. This operator is also maximal monotone. By Proposition 4, it will be enough to show that $D(T) \cap D(B_k)^0 \neq \emptyset$. Actually,

$$x^k \in D(T) \cap C^0 \cap S_k^0 = D(T) \cap D(B_k)^0.$$

Then A_k is maximal monotone. Now using Proposition 5 again we get that A_k is onto because its domain is a subset of the $D(B_k)$, which is bounded. Then there exists $y \in D(A_k) = D(T) \cap D(B_k) \subset D(T) \cap C^0$ such that

$$(28) \quad 0 \in T(y) + N_k(y) + \lambda_k \partial_x D_g(y, x^k).$$

We claim that $y \in S_k^0$. If $y = x^k$, then this is true by induction. Otherwise, let us call u^k, w^k , and v^k the elements in H such that

$$u^k \in T(y), w^k \in N_k(y), \text{ and } v^k \in \lambda_k \partial_x D_g(y, x^k),$$

and

$$(29) \quad 0 = u^k + w^k + v^k.$$

Since $y \in D(A_k) \subset D(B_k) \subset D(\partial_x D_g(\cdot, x^k)) = C^0$ by Lemma 1, it will be enough to show, in view of (27), that

$$(30) \quad D_g(y, x^k) < \frac{h(x^k)}{\lambda_k}.$$

We proceed to establish (30). Applying the gradient inequality to $D_g(\cdot, x^k)$, which is a strictly convex function, we obtain

$$(31) \quad 0 = D_g(x^k, x^k) > D_g(y, x^k) + \left\langle \frac{v^k}{\lambda_k}, x^k - y \right\rangle.$$

Using (29) in (31) and rearranging terms, since $y \neq x^k$,

$$(32) \quad \frac{1}{\lambda_k} (\langle u^k, x^k - y \rangle + \langle w^k, x^k - y \rangle) > D_g(y, x^k).$$

Since $w^k \in N_k(y)$ and $x^k \in S_k$, we have that

$$(33) \quad \langle w^k, x^k - y \rangle \leq 0.$$

Using (33) in (32) we get

$$(34) \quad 0 \leq \lambda_k D_g(y, x^k) < \langle u^k, x^k - y \rangle.$$

Since $\langle u^k, x^k - y \rangle \leq \sup_{z \in C \cap D(T), v \in Tz} \langle v, x^k - z \rangle = h(x^k)$, we obtain from (34) that $D_g(y, x^k) < \frac{h(x^k)}{\lambda_k}$; i.e., (30) holds. We have proved that $y \in S_k^0$, and therefore $N_k(y) = \{0\}$. Then $w^k = 0$, and by (29) $0 = u^k + v^k$. In view of (28), $0 \in T_k(y)$. By strict monotonicity of T_k , y is the only zero of T_k . By (18), $y = x^{k+1}$. As proved above $y \in C^0$. So $x^{k+1} \in C^0$, and the induction step is complete. $\square$

6. Convergence results. In this section we will establish the convergence properties of the sequence generated by (17)–(18). We will proceed by steps. In the next theorem we assume neither paramonotonicity of T nor norm compatibility of g (property B_6). The proof of items (a), (b), and (c) in the next theorem follows ideas contained in [11].

THEOREM 3. *Let $T : H \rightarrow \mathcal{P}(H)$ be maximal monotone with closed domain. Consider $VIP(T, C)$ where C is a closed convex subset of H . Let g be a Bregman function with zone C^0 , i.e., it satisfies B_1 – B_5 , and suppose that the following conditions hold:*

- (i) $D(T) \cap C^0 \neq \emptyset$,
- (ii) $VIP(T, C)$ has solutions,
- (iii) T is pseudomonotone,
- (iv) $\lambda_k \in (0, \tilde{\lambda}]$ for some $\tilde{\lambda} > 0$,
- (v) either
 - (v₁) g is zone coercive, or
 - (v₂) g is boundary coercive and $h_{T,C}(x) < \infty$ for all $x \in C \cap D(T)$.

Then the sequence $\{x^k\}$ generated by (17)–(18) satisfies the following:

- (a) *The sequence $\{D_g(z, x^k)\}$ is nonincreasing for all $z \in S^*$,*
- (b) *$\{x^k\}$ is bounded and has weak cluster points,*
- (c) *$\lim_{k \rightarrow \infty} D_g(x^{k+1}, x^k) = 0$,*
- (d) *if $\bar{x}$ is a weak cluster point of $\{x^k\}$, then there exists $\bar{u} \in T(\bar{x})$ such that $\langle \bar{u}, x^* - \bar{x} \rangle = 0$.*

Proof.

(a) We remember that S^* is the set of solutions of $VIP(T, C)$. We first prove that for any $z \in S^*$

$$(35) \quad D_g(z, x^{k+1}) \leq D_g(z, x^k) - D_g(x^{k+1}, x^k).$$

Note that existence of z is guaranteed by condition (ii). By Theorems 1 and 2, using Lemma 1 and (18), there exists $x^{k+1} \in C^0$ and $u^k \in T(x^{k+1})$ such that

$$(36) \quad u^k = \lambda_k(\nabla g(x^k) - \nabla g(x^{k+1})).$$

It follows easily from (7) that the following equation holds for all $y \in C$:

$$(37) \quad \langle \nabla g(x^k) - \nabla g(x^{k+1}), x^{k+1} - y \rangle = D_g(y, x^k) - D_g(y, x^{k+1}) - D_g(x^{k+1}, x^k).$$

By (36) and (37),

$$(38) \quad \langle u^k, x^{k+1} - y \rangle = \lambda_k(D_g(y, x^k) - D_g(y, x^{k+1}) - D_g(x^{k+1}, x^k)).$$

Let $z \in S^*$, and take $v^* \in T(z)$ such that

$$(39) \quad \langle v^*, x - z \rangle \geq 0 \text{ for any } x \in C.$$

By (39) and monotonicity of T ,

$$(40) \quad \langle u^k, x^{k+1} - z \rangle \geq \langle v^*, x^{k+1} - z \rangle \geq 0.$$

So (35) follows from (38) with $y = z$ and (40). By (35) and nonnegativity of D_g ,

$$(41) \quad D_g(z, x^{k+1}) \leq D_g(z, x^k),$$

which establishes (a).

(b) Iterating (41) we get $D_g(z, x^k) \leq D_g(z, x^0)$ for all k and all $z \in S^*$. Fix now some $z \in S^*$. We have shown that $\{x^k\}$ is contained in the set $\{x \in C \mid D_g(z, x) \leq D_g(z, x^0)\}$ for any $z \in S^*$, which is bounded by B_3 . So $\{x^k\}$ is bounded, and the existence of weak cluster points follows.

(c) We need to prove that

$$(42) \quad \lim_{k \rightarrow \infty} D_g(x^{k+1}, x^k) = 0.$$

By equation (35), we have that

$$(43) \quad 0 \leq D_g(x^{k+1}, x^k) \leq D_g(z, x^k) - D_g(z, x^{k+1})$$

for any $z \in S^*$. By (a), $\{D_g(z, x^k)\}$ is a nonincreasing sequence bounded below by 0 and hence convergent to some $\alpha > 0$. It follows that the limit of the rightmost expression in (43) as k goes to infinity is $\alpha - \alpha = 0$, so (42) holds.

(d) Suppose now that we have a weak cluster point of $\{x^k\}$. First of all, we observe that any weak cluster point of $\{x^k\}$ is in C . This follows from the fact that C , being closed and convex, is weakly closed. Let $\bar{x}$ be a weak cluster point and $\{x^{k_j}\}$ a subsequence of $\{x^k\}$ weakly convergent to $\bar{x}$.

In the same way as in (a), we get from (38), for $y = \bar{x}$ and $k = k_j$,

$$(44) \quad \langle u^{k_j}, x^{k_j+1} - \bar{x} \rangle = \lambda_{k_j}(D_g(\bar{x}, x^{k_j}) - D_g(\bar{x}, x^{k_j+1}) - D_g(x^{k_j+1}, x^{k_j})).$$

Now we will use conditions B_4 and B_5 to prove that the right-hand side of (44) tends to 0 as $j \rightarrow \infty$. We have that

(d1) $\{x^{k_j+1}\}$ is bounded by (a),

(d2) $x^{k_j} \xrightarrow{w} \bar{x}$ by definition of $\{x^{k_j}\}$,

(d3) $\lim_{j \rightarrow \infty} D_g(x^{k_j+1}, x^{k_j}) = 0$ by (c).

Then, applying B_5 to the sequences $\{x^{k_j+1}\}$ and $\{x^{k_j}\}$, we obtain

$$(45) \quad x^{k_j+1} \xrightarrow{w} \bar{x}.$$

In view of (45), (d2), and (d3) we can apply B_4 to the sequences $\{x^{k_j+1}\}$ and $\{x^{k_j}\}$ to obtain

$$(46) \quad \lim_{j \rightarrow \infty} (D_g(\bar{x}, x^{k_j+1}) - D_g(\bar{x}, x^{k_j})) = 0.$$

Since $\{\lambda_k\}$ is bounded above by (iv), using (d3) and (46) in (44) we get

$$(47) \quad \lim_{j \rightarrow \infty} \langle u^{k_j}, x^{k_j+1} - \bar{x} \rangle = 0.$$

Now we can use the pseudomonotonicity assumption. In fact,

$$(48) \quad x^{k_j+1} \xrightarrow{w} \bar{x} \text{ and } u^{k_j} \in Tx^{k_j+1} \text{ for all } j.$$

The pseudomonotonicity assumption, (47), and (48) imply that for any $z \in S^*$ there exists $\bar{u} \in T\bar{x}$ such that

$$(49) \quad \langle \bar{u}, \bar{x} - z \rangle \leq \liminf_{j \rightarrow \infty} \langle u^{k_j}, x^{k_j+1} - z \rangle.$$

By (38) with $y = z \in S^*$,

$$(50) \quad \langle u^k, x^{k+1} - z \rangle = \lambda_k (D_g(z, x^k) - D_g(z, x^{k+1}) - D_g(x^{k+1}, x^k)).$$

By (a), the sequence $D_g(z, x^k)$ is decreasing and then the right-hand side of (50) tends to zero because $\{\lambda_k\}$ is bounded above by (iv). So the limit of the right side of (49) is 0. Taking limits in (49) as $j \rightarrow \infty$,

$$(51) \quad \langle \bar{u}, \bar{x} - z \rangle \leq 0$$

with $\bar{u} \in T(\bar{x})$. Since $z \in S^*$, there exists $v^* \in T(z)$ such that $\langle v^*, y - z \rangle \geq 0$ for all $y \in C$. Therefore, by monotonicity of T ,

$$(52) \quad \langle \bar{u}, \bar{x} - z \rangle \geq \langle v^*, \bar{x} - z \rangle \geq 0.$$

(51) and (52) imply that $\langle \bar{u}, \bar{x} - z \rangle = 0$ and (d) is established. $\square$

Now, assuming that g is norm compatible and that T is paramonotone in C , we prove that the sequence generated by GPPA is weakly convergent to a solution of $VIP(T, C)$.

THEOREM 4. *Let $T : H \rightarrow \mathcal{P}(H)$ be maximal monotone. Consider $VIP(T, C)$, where C is a closed convex subset of H . Let g be a Bregman function with zone C^0 , i.e., it satisfies B_1 – B_5 , and suppose that the following conditions hold:*

- (i) $D(T) \cap C^0 \neq \emptyset$,
- (ii) $S^* \neq \emptyset$,
- (iii) T is pseudomonotone from $D(T)$ to H ,
- (iv) $\lambda_k \in (0, \tilde{\lambda}]$ for some $\tilde{\lambda} > 0$,
- (v) either
 - (v₁) g is zone coercive, or
 - (v₂) g is boundary coercive, and $h_{T,C}(x) < \infty$ for all $x \in C \cap D(T)$,

- (vi) g is norm compatible (i.e., B_8 holds),
- (vii) T is paramonotone in C .

Then the sequence $\{x^k\}$ given by (17)–(18) is weakly convergent to a solution $\bar{x}$ of $VIP(T, C)$.

Proof. By Theorem 3(b), the sequence $\{x^k\}$ has weak cluster points. By (vii), Proposition 1(ii), and Theorem 3(d), all weak cluster points of $\{x^k\}$ belong to S^* . We will prove that if g is norm compatible, then there exists only one weak cluster point. By Theorem 3(a), for any fixed $z \in S^*$, we have that the sequence $\beta_k(z) := \langle \nabla g(x^k), x^k - z \rangle - g(z)$ is decreasing and $\beta_k(z) \geq -g(z)$ because $D_g(z, x^k) \geq 0$. Then $\{\beta_k(z)\}$ is convergent. Suppose that there are at least two different weak cluster points $z_1, z_2 \in S^*$. Take subsequences $\{x^{j_k}\}, \{x^{i_k}\}$ of $\{x^k\}$ such that $x^{j_k} \xrightarrow{w} z_1, x^{i_k} \xrightarrow{w} z_2$. By convergence of the sequences $\{\beta_k(z_1)\}$ and $\{\beta_k(z_2)\}$, there exist $\ell_1, \ell_2 \in \mathbf{R}$ such that

$$\lim_{k \rightarrow \infty} \beta_k(z_1) = \ell_1, \quad \lim_{k \rightarrow \infty} \beta_k(z_2) = \ell_2.$$

Then

$$(53) \quad \lim_{k \rightarrow \infty} (\beta_k(z_1) - \beta_k(z_2)) = \lim_{k \rightarrow \infty} \langle \nabla g(x^k), z_2 - z_1 \rangle = \ell_1 - \ell_2.$$

Considering the subsequences $\{x^{j_k}\}$ and $\{x^{i_k}\}$ in (53), we obtain

$$(54) \quad \lim_{k \rightarrow \infty} \langle \nabla g(x^{j_k}) - \nabla g(x^{i_k}), z_2 - z_1 \rangle = 0.$$

Since $z_1 \neq z_2$, (54) contradicts the assumption of norm compatibility on g . So there is only one weak cluster point of $\{x^k\}$ and the whole sequence converges weakly to it. $\square$

In the case when T is paramonotone, but we do not require g to be norm compatible, we obtain the following result.

COROLLARY 1. *Suppose that all the conditions of Theorem 3 hold. Assume furthermore that T is paramonotone in C . Then the sequence $\{x^k\}$ generated by (17)–(18) is bounded, $\lim_{k \rightarrow \infty} D_g(x^{k+1}, x^k) = 0$, and all weak cluster points of $\{x^k\}$ are solutions of $VIP(T, C)$.*

Proof. The proof follows from Theorem 3 and Proposition 1(ii). $\square$

We point out that the pseudomonotonicity assumption required in Theorems 3 and 4, and in Corollary 1, is guaranteed to hold in the cases considered in Proposition 2.

We close this section showing that existence of solutions of $VIP(T, C)$ is a necessary condition for convergence of the sequence $\{x^k\}$ generated by (17)–(18). In fact, we will show that if the solution set S^* of $VIP(T, C)$ is empty, then $\{x^k\}$ is unbounded. We need a preparatory result.

PROPOSITION 6. *If V is a closed and convex set, then the normality operator N_V is pseudomonotone from $D(N_V)$.*

Proof. Take $\{x^k\}, \{w^k\}, x^0$, and y as in Definition 3. Since $x^k, y \in D(N_V) = V$, it follows easily from the definition of N_V that $\langle w^k, x^k - y \rangle \geq 0$. On the other hand, since $N_V(x^0)$ is a positive cone, it holds that $0 \in N_V(x^0)$. Take $w^0 = 0$ and conclude that $\langle w^0, x^0 - y \rangle = 0 \leq \liminf_{k \rightarrow \infty} \langle w^k, x^k - y \rangle$. $\square$

LEMMA 6. *Under the hypotheses of Corollary 1, let $\{x^k\}$ be the sequence generated by (17)–(18). If $S^* = \emptyset$, then $\{x^k\}$ is unbounded.*

Proof. Suppose that $\{x^k\}$ is bounded. Then there exists a closed, bounded, and convex set $B \subset H$ such that $\overline{\{x^k\}} \subset B^0$. Consider the operator $\tilde{T} := T + N_B$, where N_B is the normality operator associated with B . Remember that $N_B(x) = \{0\}$ for all $x \in B^0$. Let $\{\tilde{x}^k\}$ be the sequence generated by (17)–(18) for the operator $\tilde{T}$ with $\tilde{x}^0 = x^0$. We will prove by induction that $x^k = \tilde{x}^k$ for all k . The result holds for $k = 0$ by assumption. Suppose that $x^k = \tilde{x}^k$. By (18),

$$0 \in T(x^{k+1}) + \lambda_k \partial_x D_g(x^{k+1}, x^k) = \tilde{T}(x^{k+1}) + \lambda_k \partial_x D_g(x^{k+1}, \tilde{x}^k).$$

using the fact that $T = \tilde{T}$ in B^0 and the inductive hypothesis. Then x^{k+1} is a solution of the equation in y ,

$$(55) \quad 0 \in \tilde{T}(y) + \lambda_k \partial_x D_g(y, x^k).$$

By strict monotonicity of the operator $\tilde{T} + \lambda_k \nabla_x D_g(\cdot, x^k)$, the solution y of (55) is unique. Since by (18) $\tilde{x}^{k+1}$ is the solution of (55), we conclude that $\tilde{x}^{k+1} = x^{k+1}$. The induction step is complete.

We check now the hypotheses of Corollary 1 for $VIP(\tilde{T}, C)$. It is easy to verify, using Proposition 4, that $\tilde{T}$ is maximal monotone. Since $\tilde{T} = T + \partial(\delta_B)$, $\tilde{T}$ is paramonotone by Proposition 1(i) and (iii). It follows from Proposition 6 that N_B is pseudomonotone. So $\tilde{T}$ is the sum of two pseudomonotone operators, and therefore it is pseudomonotone (see [19, p. 97]). Finally, we verify that $VIP(\tilde{T}, C)$ has solutions. As mentioned above, the set of solutions of $VIP(\tilde{T}, C)$ is the set of zeroes of $\tilde{T} + N_C = T + N_B + N_C$. Note that $D(\tilde{T} + N_C) = D(T) \cap C \cap B \subset B$. Since B is bounded, $\tilde{T} + N_C$ is onto by Proposition 5, and therefore it has zeroes. We can apply Corollary 1 and conclude that $\{\tilde{x}^k\}$, and therefore $\{x^k\}$, has weak cluster points and all of them are solutions of $VIP(\tilde{T}, C)$. Let x^* be a cluster point of $\{x^k\}$. It follows from our assumption on $\{x^k\}$ that $x^* \in B^0$. Since $T(x) = \tilde{T}(x)$ for all $x \in B^0$, we get $T(x^*) = \tilde{T}(x^*)$, and then it follows from (1) that x^* is also a solution of $VIP(T, C)$, in contradiction with our hypothesis. We conclude that $\{x^k\}$ is unbounded. $\square$

We summarize our results in the following theorem.

THEOREM 5. *Assume that the hypotheses of Theorem 4 hold. Let $\{x^k\}$ be the sequence generated by (17)–(18). Then*

- (i) $\{x^k\}$ is weakly convergent if and only if $S^* \neq \emptyset$.
- (ii) If $S^* \neq \emptyset$, then the weak limit x^* of $\{x^k\}$ belongs to S^* .
- (iii) If $S^* = \emptyset$, then $\{x^k\}$ is unbounded.

Proof.

(i) If $S^* \neq \emptyset$, then the result follows from Theorem 4. If the sequence converges, then it is bounded and it follows from Lemma 6 that S^* is nonempty.

(ii) Follows from Theorem 4.

(iii) Follows from Lemma 6. $\square$

7. The finite-dimensional case. We state in this section our results adapted to the finite-dimensional case. Some of the conditions become simpler in this situation. The weak convergence is replaced by the strong one, and the norm compatibility of the Bregman function holds automatically. The conditions in the definition of Bregman function simplify, and we obtain a slight improvement in item (c) of Theorem 3.

We will say that g is a *Bregman function* if it satisfies conditions B_1^* – B_5^* below. D_g is still defined by (7), where now ∇g denotes the usual gradient of g .

(B_1^*) g is continuously differentiable in S and continuous in $\bar{S}$.

(B_2^*) g is strictly convex in $\bar{S}$.

(B₃^{*}) For all $\delta \in \mathbf{R}$, the partial level set $\Gamma_2(x, \delta) = \{y \in S \mid D_g(x, y) \leq \delta\}$ is bounded for all $x \in \bar{S}$.

(B₄^{*}) If $\{y^k\} \subset S$ converges to $\hat{y}$, then $D_g(\hat{y}, y^k)$ converges to 0.

(B₅^{*}) If $\{x^k\} \subset \bar{S}$ and $\{z^k\} \subset S$ are sequences such that $\{x^k\}$ is bounded, $\lim_{k \rightarrow \infty} z^k = \hat{z}$, and $\lim_{k \rightarrow \infty} D_g(x^k, z^k) = 0$, then $\lim_{k \rightarrow \infty} x^k = \hat{z}$.

It is easy to check that properties B₄^{*} and B₅^{*} hold automatically as a consequence of B₁^{*}–B₃^{*} when $\hat{y}$, $\{x^k\}$, and $\hat{z}$ are in S and so they need to be checked only in the boundary ∂S of S . We prove next that in finite dimension, B₁^{*}–B₅^{*} imply B₁–B₆.

PROPOSITION 7. *If H is finite dimensional and g satisfies B₁^{*}–B₅^{*}, then g satisfies B₁–B₆.*

Proof. Conditions B₁ and B₂ are straightforward consequences of B₁^{*} and B₂^{*}. In order to verify B₃, it suffices to check that $\Gamma_1(y, \delta)$ is bounded. Note that B₂^{*} implies that $D_g(\cdot, y)$ is strictly convex so that $\Gamma_1(y, 0) = \{y\}$, which is a bounded set. In finite dimension, if a convex function has one bounded level set, then all its level sets are bounded; i.e., $\Gamma_1(y, \delta)$ is bounded for all δ (note that this argument fails in infinite dimension so that boundedness of $\Gamma_1(y, \delta)$ must be specifically required in B₃).

We proceed to verify B₄. Given two sequences $\{x^k\}, \{y^k\} \subset S$, such that $x^k \xrightarrow{w} x$, $y^k \xrightarrow{w} x$, and $\lim_{k \rightarrow \infty} D_g(x^k, y^k) = 0$, we have to prove that $\lim_{k \rightarrow \infty} [D_g(x, x^k) - D_g(x, y^k)] = 0$. By B₄^{*}, $\lim_{k \rightarrow \infty} D_g(x, x^k) = \lim_{k \rightarrow \infty} D_g(x, y^k) = 0$, which establishes the result.

Condition B₅^{*} implies B₅ trivially because in this case the notions of strong and weak convergence coincide.

The only property that requires some work is B₆. Suppose that $\{x^k\} \subset C^0$ and $\{y^k\} \subset C^0$ are such that $\lim_{k \rightarrow \infty} x^k = x$ and $\lim_{k \rightarrow \infty} y^k = y$. We will prove that if

$$(56) \quad \liminf_k |\langle \nabla g(x^k) - \nabla g(y^k), x - y \rangle| = 0,$$

then it holds that $x = y$. This establishes condition B₆. After some algebra, we get from (7) that

$$(57) \quad \langle \nabla g(x^k) - \nabla g(y^k), x - y \rangle = D_g(x, y^k) + D_g(y, x^k) - D_g(x, x^k) - D_g(y, y^k).$$

By B₄^{*}, we have that the last two terms in the right-hand side of (57) tend to zero. So under hypothesis (56) it follows from (57) that $\lim_{k \rightarrow \infty} D_g(x, y^k) = \lim_{k \rightarrow \infty} D_g(y, x^k) = 0$ using the nonnegativity of D_g . Since $\lim_{k \rightarrow \infty} y^k = y$ by (i), if we consider the sequence $\{z^k\}$, defined as $z^k := x$ for all k , we can apply B₅^{*} to the sequences $\{z^k\}$ and $\{y^k\}$ and conclude that $x = y$. So B₆ is established. $\square$

We present now our convergence result in the finite-dimensional case.

COROLLARY 2. *Let $T : H \rightarrow \mathcal{P}(H)$ be maximal monotone. Consider $VIP(T, C)$, where C is a closed convex subset of $\mathbf{R}^n$. Let g be a Bregman function with zone C^0 , i.e., it satisfies B₁^{*}–B₅^{*}, and suppose that the following conditions hold:*

- (i) $D(T) \cap C^0 \neq \emptyset$,
- (ii) $VIP(T, C)$ has solutions,
- (iii) T is pseudomonotone from $D(T)$ to $\mathcal{P}(\mathbf{R}^n)$,
- (iv) $\lambda_k \in (0, \tilde{\lambda}]$ for some $\tilde{\lambda} > 0$,
- (v) either

(v₁) g is zone coercive, or

(v₂) g is boundary coercive, and $h_{T,C}(x) < \infty$ for all $x \in C \cap D(T)$.

Then the sequence $\{x^k\}$ generated by (17)–(18) satisfies the following:

- (a) *The sequence $\{D_g(z, x^k)\}$ is nonincreasing for all $z \in S^*$,*

- (b) $\{x^k\}$ is bounded and has weak cluster points,
 - (c) $\lim_{k \rightarrow \infty} (x^{k+1} - x^k) = 0$,
 - (d) if $\bar{x}$ is a cluster point of $\{x^k\}$, then there exists $\bar{u} \in T(\bar{x})$ such that $\langle \bar{u}, x^* - \bar{x} \rangle = 0$,
 - (e) if a cluster point x^* of $\{x^k\}$ solves $VIP(T, C)$, then $x^* = \lim_{k \rightarrow \infty} x^k$.
- If in addition T is paramonotone in C , then
- (f) $\{x^k\}$ converges to a solution $\bar{x}$ of $VIP(T, C)$.

Proof. By Proposition 7, B_1 – B_5 hold. Since in finite dimension the notions of weak and strong convergence coincide, items (a), (b), and (d) follow from Theorem 3(a), (b), and (d), and we also have $\lim_{k \rightarrow \infty} D_g(x^{k+1}, x^k) = 0$ by Theorem 3(c). We claim that in finite dimension this implies $\lim_{k \rightarrow \infty} (x^{k+1} - x^k) = 0$. Otherwise, using (b), there exists a subsequence $\{x^{\ell_k}\}$ of $\{x^k\}$ such that $\lim_{k \rightarrow \infty} x^{\ell_k} = \tilde{x} \neq \hat{x} = \lim_{k \rightarrow \infty} x^{\ell_k+1}$, but $\lim_{k \rightarrow \infty} D_g(x^{k+1}, x^k) = 0$ and B_5^* imply $\tilde{x} = \hat{x}$. The claim is established and (c) holds. We proceed to prove (e). By (a), $\{D_g(x^*, x^k)\}$ is nonincreasing and also nonnegative. Let $\{x^{i_k}\}$ be a subsequence of $\{x^k\}$ such that $\lim_{k \rightarrow \infty} x^{i_k} = x^*$. By B_4^* , $\lim_{k \rightarrow \infty} D_g(x^*, x^{i_k}) = 0$. Then $\{D_g(x^*, x^k)\}$ is a nonincreasing and nonnegative sequence which has a subsequence converging to 0. It follows that the whole sequence converges to 0, and from $\lim_{k \rightarrow \infty} D_g(x^*, x^k) = 0$ we get, using B_5^* , that $x^* = \lim_{k \rightarrow \infty} x^k$. For (f) we apply Theorem 4, noting that norm compatibility holds by Proposition 7. $\square$

In this case also the pseudomonotonicity assumption of Corollary 2 is known to hold in the situations considered in Proposition 2.

REFERENCES

- [1] L. M. BREGMAN, *The relaxation method of finding the common points of convex sets and its application to the solution of problems in convex programming*, USSR Computational Mathematics and Mathematical Physics, 7 (1967), pp. 200–217.
- [2] H. BRÉZIS AND A. HARAUX, *Image d'une somme d'opérateurs monotones et applications*, Israel J. Math., 23 (1976), pp. 165–186.
- [3] H. BRÉZIS, *Opérateurs Monotones Maximaux et Semigroups de Contractions dans les Espaces de Hilbert*, Mathematics Studies 5, North-Holland, New York, 1973.
- [4] F. E. BROWDER, *Nonlinear Operators and Nonlinear Equations of Evolution in Banach Spaces*, Proc. Sympos. Pure Math. 18, American Mathematical Society, Providence, RI, 1976.
- [5] R. S. BURACHIK, *Generalized Proximal Point Methods for the Variational Inequality Problem*, Ph.D. thesis, Instituto de Matemática Pura e Aplicada, Rio de Janeiro, 1995.
- [6] Y. CENSOR AND S. ZENIOS, *The proximal minimization algorithm with D-functions*, J. Optim. Theory Appl., 73 (1992), pp. 451–464.
- [7] Y. CENSOR, A. N. IUSEM, AND S. A. ZENIOS, *An interior point method with Bregman functions for the variational inequality problem with paramonotone operators*, Math. Programming, to appear.
- [8] G. CHEN AND M. TEBoulLE, *Convergence analysis of a proximal-like minimization algorithm using Bregman functions*, SIAM J. Optim., 3 (1993), pp. 538–543.
- [9] J. ECKSTEIN, *Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming*, Math. Oper. Res., 18 (1993), pp. 202–226.
- [10] P. P. B. EGGERMONT, *Multiplicative iterative algorithms for convex programming*, Linear Algebra Appl., 130 (1990), pp. 25–42.
- [11] O. GÜLER, *On the convergence of the proximal point algorithm for convex minimization*, SIAM J. Control Optim., 29 (1991), pp. 403–419.
- [12] P. T. HARKER AND J. S. PANG, *Finite dimensional variational inequalities and nonlinear complementarity problems: A survey of theory, algorithms and applications*, Math. Programming, 48 (1990), pp. 161–220.
- [13] A. N. IUSEM, *On some properties of generalized proximal point methods for quadratic and linear programming*, J. Optim. Theory Appl., 85 (1995), pp. 513–612.
- [14] A. N. IUSEM, *On some properties of paramonotone operators*, manuscript.

- [15] S. KARAMARDIAN, *Complementarity over cones with monotone and pseudomonotone maps*, J. Optim. Theory Appl., 18 (1976), pp. 445–454.
- [16] D. KINDERLEHRER AND G. STAMPACCHIA, *An Introduction to Variational Inequalities and Their Applications*, Academic Press, New York, 1980.
- [17] K. KIWIEL, *Proximal minimization methods with generalized Bregman functions*, SIAM J. Control Optim., 35 (1997), pp. 1142–1168.
- [18] B. LEMAIRE, *The proximal algorithm*, in Internat. Ser. Numer. Math., J. P. Penot, ed., Birkhäuser, Basel, 87 (1989), pp. 73–87.
- [19] D. PASCALI AND S. SBURLAN, *Nonlinear Mappings of Monotone Type*, Editura Academiei, Bucharest, 1978.
- [20] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
- [21] R. T. ROCKAFELLAR, *On the maximality of sums of nonlinear monotone operators*, Trans. Amer. Math. Soc., 149 (1970), pp. 75–88.
- [22] R. T. ROCKAFELLAR, *Monotone operators and the proximal point algorithm*, SIAM J. Control Optim., 14 (1976), pp. 877–898.

BUDGET-DEPENDENT CONVERGENCE RATE OF STOCHASTIC APPROXIMATION*

PIERRE L'ECUYER[†] AND GEORGE YIN[‡]

Abstract. Convergence rate results are derived for a stochastic optimization problem where a performance measure is minimized with respect to a vector parameter θ . Assuming that a gradient estimator is available and that both the bias and the variance of the estimator are (known) functions of the budget devoted to its computation, the gradient estimator is employed in conjunction with a stochastic approximation (SA) algorithm. Our interest is to figure out how to allocate the total available computational budget to the successive SA iterations. The effort is devoted to solving the asymptotic version of this problem by finding the convergence rate of SA toward the optimizer, first as a function of the number of iterations and then as a function of the total computational effort. As a result the optimal rate of increase of the computational budget per iteration can be found. Explicit expressions for the case where the computational budget devoted to an iteration is a polynomial in the iteration number, and where the bias and variance of the gradient estimator are polynomials of the computational budget, are derived. Applications include the optimization of steady-state simulation models with likelihood ratio, perturbation analysis, or finite-difference gradient estimators; optimization of infinite-horizon models with discounting; optimization of functions of several expectations; and so on. Several examples are discussed. Our results readily generalize to general root-finding problems.

Key words. stochastic optimization, discrete-event systems, stochastic approximation, gradient estimate, rate of convergence, limit theorems

AMS subject classifications. 60F05, 60F17, 62L20, 93E23, 93E25, 93E30

PII. S1052623495270723

1. Introduction, motivation, and examples. Consider a stochastic model parameterized by a vector $\theta \in G \subseteq \mathbb{R}^r$, where G is either the entire space $\mathbb{R}^r$ or a closed and convex set. The objective is to minimize a cost functional $\alpha(\cdot)$ over G , assuming that α has a unique minimizer θ^* in the interior of G . We are mainly interested in the situation where the exact value of $\alpha(\theta)$ (or its gradient) is too hard to compute and where only a gradient estimator is available, which can be computed, say, by simulation. The quality of the estimator (e.g., bias, variance, etc.) might depend on the parameter value θ at which it is evaluated and also on the computing budget that we are ready to spend to perform the simulation. To simplify the notation, we assume henceforth that $G = \mathbb{R}^r$. Algorithms with projections, which we use when $G \neq \mathbb{R}^r$, are discussed at the end of the paper. Their asymptotic properties are similar, provided that θ^* is in the interior of G . We do not consider the more complicated situation where the optimum lies on the boundary of G ; it requires different techniques for obtaining the rates of convergence. Although we concentrate on the stochastic optimization problems, for which we are finding a root of the gradient of α , the results obtained are also applicable to more general root-finding problems under noisy measurements or observations.

* Received by the editors March 21, 1995; accepted for publication (in revised form) January 6, 1997.

<http://www.siam.org/journals/siopt/8-1/27072.html>

[†] Département d'I.R.O., Université de Montréal, C.P. 6128, Succ. Centre-Ville, Montréal, H3C 3J7, Canada (lecuyer@iro.umontreal.ca). This author was supported by NSERC-Canada grant OGP0110050 and FCAR-Québec grant EQ2831.

[‡] Department of Mathematics, Wayne State University, Detroit, MI 48202 (gyin@math.wayne.edu). This author was supported in part by the National Science Foundation under grants DMS-9529738 and DMS-9600245 and in part by Wayne State University.

We consider the SA algorithm

$$(1.1) \quad \theta_{n+1} = \theta_n - a_n \psi_n$$

for $n \geq 1$, where $\theta_1 \in G$ is the initial parameter estimate, ψ_n is the value of a gradient estimator (at parameter value θ_n) obtained at iteration n , and $\{a_n, n \geq 1\}$ is a decreasing sequence such that $a_n > 0$, $\sum_n a_n = \infty$, and $\sum_n a_n^2 < \infty$. Let T_n be the size of the computing effort made to compute ψ_n . Here, ψ_n and T_n are (generally correlated) random variables whose distributions depend in general on the parameter value θ_n , on the initial conditions (or state) of the simulation at step n (say, s_n), on the simulation time-horizon and/or number of replications or number of regenerative cycles at step n , and perhaps on other factors, such as the size of the finite differences used at step n in the case where a finite-difference (FD) gradient estimator is used. We assume that the probability distribution of (ψ_n, T_n, s_{n+1}) depends on (n, s_n, θ_n) and that conditional on (n, s_n, θ_n) , that distribution is independent of the past. In other words, $\{(n, \psi_n, T_n, s_{n+1}, \theta_{n+1}), n \geq 1\}$ evolves as a Markov chain. In some situations, T_n could be a deterministic computing budget devoted to iteration n (its distribution is then degenerate), while in other cases the number of replications is fixed and T_n is random (although we have some indirect control over it). This formulation is quite general. Denote by E_n the expectation conditional on (n, s_n, θ_n) .

Let $\alpha_\theta(\cdot)$ denote the gradient of $\alpha(\cdot)$. We define the conditional bias and variance of the gradient estimator at step n as

$$(1.2) \quad B_n = E_n[\psi_n] - \alpha_\theta(\theta_n)$$

and

$$(1.3) \quad V_n = E[\|\psi_n - E_n \psi_n\|^2],$$

respectively, where $\|\cdot\|$ denotes the Euclidean norm. B_n is a random variable, but not V_n . Note that V_n is the trace of the “covariance” matrix $E[(\psi_n - E_n \psi_n)(\psi_n - E_n \psi_n)']$ (here and throughout the paper, the prime denotes matrix transpose). Note that in V_n , E_n is used in lieu of E as in the standard definition of variance, but for convenience, we still call it variance. We shall use the following notation: If f and g are positive-valued functions defined over the natural numbers, we say that “ $f(n)$ is of the order $g(n)$ as $n \rightarrow \infty$,” denoted by $f(n) = O(g(n))$, if there exists a positive integer n_0 and a constant $K > 0$ such that $f(n) \leq Kg(n)$ for all $n \geq n_0$. If $f(n) = O(g(n))$ and $g(n) = O(f(n))$, we say that “ $f(n)$ is of the exact order $g(n)$,” denoted by $f(n) = \Theta(g(n))$. If $f(n)/g(n) \rightarrow 1$, we say that “ $f(n)$ is similar to $g(n)$,” denoted by $f(n) \sim g(n)$.

SA algorithms have been studied extensively and employed in a wide range of applications; see, for example, Wasan [40], Nevel’son and Khasminskii [32], Kushner [19], Kushner and Clark [20], and L’Ecuyer and Glynn [29], among others. For an extensive survey on general SA for parameterized models of either continuous- or discrete-event systems, see Kushner and Vázquez-Abad [22] and the references therein. Those analyses generally assume that the computational effort T_n is (roughly) the same at all iterations, and the convergence rates are obtained in terms of the number of iterations only.

In this paper, we consider a more general case where the computing budget could vary between iterations and analyze the convergence rate of the mean square error

(MSE), defined as $E\|\theta_n - \theta^*\|^2$, in terms of the total computational effort

$$(1.4) \quad C_n = \sum_{i=1}^n T_i.$$

We seek to achieve

$$(1.5) \quad E[\|\theta_n - \theta^*\|^2] = O((E[C_n])^{-\eta})$$

for the largest possible constant η . This is motivated by the fact that in certain situations (see the examples below), B_n will converge to zero (and θ_n will converge to the optimizer) only if the computing time T_n increases towards infinity as a function of n , thus the T_n 's are not the same at all the iterations (and are also typically random). In that context, the convergence rate expressed as a function of the total computational effort is more meaningful than that expressed as a function of n . Typically, both B_n and V_n are (roughly) polynomial functions of $E[T_n]$ and the rate of change of the latter can be chosen directly or indirectly. An interesting question is then: What is the optimal rate of increase of $E[T_n]$? A first step toward answering this question is to figure out how the MSE depends on B_n and V_n .

Glynn and Whitt [16] have developed a framework for studying the asymptotic efficiency of simulation estimators as a function of the available computational budget. Their goal was to capture the interplay between the variability of an estimator and the computational effort required. They obtained limit theorems for several examples including two SA settings: classical Robbins–Monro and Kiefer–Wolfowitz based on central finite differences. Their analysis for these two examples is based on convergence rate results (in terms of n) by Ruppert [36]. Our results generalize those studies. To further motivate our development, we now introduce some examples where our framework is appropriate.

Example 1.1. Consider a $GI/GI/1$ queue with mean arrival rate $\lambda = 1$ and mean service time $\theta \geq 0$, such that θ is a scale parameter of the service time distribution. More specifically, each service time can be written as $S = \theta Z$ where Z has mean 1 and a distribution independent of θ . For $0 \leq \theta < 1$, the system is stable. Let $w(\theta)$ be the steady-state average wait (in the queue) per customer. Suppose that we want to minimize the function $\alpha(\theta) = w(\theta) + C(\theta)$, where C is a smooth convex function with known derivative, over the interval $G = [0, \bar{\theta}]$, for some $0 < \bar{\theta} < 1$. We design the algorithm so that the iterate is projected back to $G = [0, \bar{\theta}]$ whenever it goes outside, and we take $\bar{\theta} < 1$ to make sure that the system is stable for each θ considered (in fact, this is necessary if one simulates a given number of busy cycles at each step of the SA algorithm, but not if one simulates a given number of customers at each step). Here, to estimate the derivative of w , one can start the queue empty and simulate the system until the t th customer starts its service, or until the end of the t th busy cycle, and compute a (generally biased) derivative estimator from that. Then, one adds $dC(\theta)/d\theta$ to obtain a derivative estimator for α . Several specific estimators for the derivative of $w(\theta)$, based on a finite-horizon simulation, are discussed and experimented in L'Ecuyer and Glynn [29] and L'Ecuyer, Giroux, and Glynn [28]. Generally, those estimators have a bias of $O(t^{-1})$ due to the truncation of the horizon and sometimes additional bias such as that due to the use of finite differences to estimate the gradient. Their variances have different orders of magnitudes. For example, to mention a few, three estimators called infinitesimal perturbation analysis (IPA), likelihood ratio (LR), and LR with control variates (CLR), all based on the simulation of the first t customers,

have variances of $O(t^{-1})$, $O(t)$, and $O(1)$, respectively. Here, the computing cost T_n may be assumed roughly proportional to the number of customers simulated (the value of t) at iteration n of SA. A similar situation occurs for other objective functions, more general queueing networks, and many other Markov chains or discrete-event models. For other specific examples, see Yin, Yan, and Lou [43] or Haurie, L'Ecuyer, and Van Delft [17], where IPA gradient estimators for the performance of an unreliable manufacturing system, with respect to threshold (hedging point) values, are used for optimization; these estimators have bias and variance both of $O(t^{-1})$. In those references, weak and with probability one (w.p.1) convergences were proved, but no convergence rate was obtained.

Example 1.2. Suppose that $\alpha(\theta)$ is expressed as a (differentiable) nonlinear function of one or more mathematical expectations, say, $\alpha(\theta) = g(\mu_1(\theta), \dots, \mu_d(\theta))$, where $g : \mathbb{R}^d \rightarrow \mathbb{R}$ and $\mu_j(\theta) = E[X_j]$ for some random variable X_j whose distribution depends on θ . Suppose that an unbiased estimator Y_j is available for $\mu_{j,\theta}(\theta)$, the gradient of μ_j with respect to θ , which is assumed to exist. Let $X_{j,1}, \dots, X_{j,N}$ and $Y_{j,1}, \dots, Y_{j,N}$ be N independently and identically distributed (i.i.d.) copies of X_j and Y_j , respectively, and let $\bar{X}_j(N) = (1/N) \sum_{i=1}^N X_{j,i}$ and $\bar{Y}_j(N) = (1/N) \sum_{i=1}^N Y_{j,i}$ be their sample averages. A consistent estimator for the gradient of α is then

$$\psi = \sum_{j=1}^d g_j(\bar{X}_1(N), \dots, \bar{X}_d(N)) \cdot \bar{Y}_j(N),$$

where g_j denotes the derivative of g with respect to its j th parameter. This estimator is generally biased because of the nonlinearity of g . Under appropriate conditions on g and on the estimators X_j and Y_j , it can be proved that both the bias and variance of ψ are of $\Theta(N^{-1})$ (see Glynn and Heidelberger [13]). Let $T(N)$ be the computing time for performing the required simulations and computing ψ . If $E[T(N)]/N$ converges to some positive constant as $N \rightarrow \infty$, uniformly in θ , then $E[T_n]E[B_n]$ and $E[T_n]V_n$ are $\Theta(1)$.

One special case of this example is when g is (or involves) the ratio of two expectations: $g(\mu_1, \mu_2) = \mu_1/\mu_2$. This case arises for example when the regenerative method is applied to construct steady-state estimators or when estimating conditional probabilities is desired, as well as in other practical situations. The gradient estimator is then

$$\psi = \frac{\bar{Y}_1(N)\bar{X}_2(N) - \bar{X}_1(N)\bar{Y}_2(N)}{(\bar{X}_2(N))^2}.$$

Gradient estimators for ratios of expectations are studied in Glynn, L'Ecuyer, and Adès [15].

Example 1.3. Consider a continuous-time stochastic process $\{Z(t), t \geq 0\}$ whose probability law depends on θ and which represents a time-varying cost rate. Let $\rho > 0$ be the discount rate and suppose that we want to minimize the total expected discounted cost

$$\alpha(\theta) = E \left[\int_0^\infty e^{-\rho t} Z(t) dt \right],$$

with respect to θ , where the expectation depends on θ . An unbiased estimator of α_θ might not be available, but a biased estimator can generally be obtained by simulating the process over a truncated (finite) horizon t using either finite differences or

perturbation analysis, or perhaps other methods. Fox and Glynn [8] analyze different estimators of $\alpha(\theta)$ in terms of their convergence rates as a function of the computational budget. They show in particular that if the cost rate is nonnegative, then the truncated-horizon estimator of $\alpha(\theta)$ has bias $O(e^{-\rho t})$ and bounded variance. When estimating the gradient, the bias and variance would also be functions of the horizon t , depending on the gradient-estimation method. The optimal choice of t as a function of n should then depend on the actual forms of these functions.

In the preceding examples, conditional on n and θ_n , the gradient estimator ψ_n is independent of s_n . In Example 1.1, the condition is satisfied because we restarted the system empty at every iteration. However, one can instead restart the queue at iteration n from the same state as it was at the end of iteration $n - 1$. In that case, ψ_n depends also on s_n , and this is covered by our results.

The rest of the paper is arranged as follows. In section 2, we give the precise formulation of the problem, state our assumptions, and present a preliminary result saying that convergence w.p.1 to θ^* occurs under our assumptions. Section 3 is devoted to obtaining a bound on the order of magnitude of the MSE. We first obtain such a bound as a function of n . Roughly, we show that if $\|B_n\| = O(n^{-\beta})$ w.p.1, $V_n = O(n^{-\delta})$ and $a_n = n^{-\gamma}$, for some real-valued constants β , δ , and $1/2 < \gamma \leq 1$, and if some other technical conditions on the shape of α are satisfied, then

$$(1.6) \quad E\|\theta_n - \theta^*\|^2 = O(n^{-\kappa}),$$

where

$$(1.7) \quad \kappa = (2\beta) \wedge (\gamma + \delta)$$

and $x \wedge y$ denotes $\min\{x, y\}$. Later on, we express that bound as a function of $E[C_n]$ for the case where $E[T_n]$ grows polynomially fast and where B_n and V_n grow at specific rates as functions of $E[T_n]$.

Section 4 deals with the asymptotic distribution of $n^{\kappa/2}(\theta_n - \theta^*)$, where κ is a constant that gives the appropriate scaling for that distribution to be nontrivial. We show that the asymptotic distribution depends on β and δ ; i.e., different decreasing rates for the bias or variance result in different scalings in the asymptotic distribution. Additional conditions to those of section 2 are required for that analysis. These conditions require in particular that $n^\beta\|B_n\|$ and $n^\delta V_n$ converge to a constant scalar and vector, respectively, and our results imply that $n^{\kappa/2}(\theta_n - \theta^*)$ either converges to a constant or to a normal random variable. In section 5, we derive the (asymptotically) optimal growth rates of T_n for several practical situations, including variants of the previous examples. As an illustration, if $(E[T_n])^b\|B_n\| = \Theta(1)$, $(E[T_n])^d V_n = \Theta(1)$, $n^{-p}E[T_n] = \Theta(1)$ for some constants $b \geq 0$, $d, p \geq 0$, and under some additional assumptions, one has

$$(1.8) \quad E\|\theta_n - \theta^*\|^2 = \Theta((E[C_n])^{-((2\beta) \wedge (\gamma + \delta))/(p+1)}) = \Theta((E[C_n])^{-\eta})$$

with $\beta = bp$ and $\delta = dp$. From that, one can then find the optimal values of p and γ , i.e., those that maximize η in (1.8), for any given values of b and d . Note that if there is no bias ($\beta = b = \infty$) and $1/2 < \gamma \leq 1$, then it makes sense to have $E[T_n]$ bounded ($p = 0$), which yields $\delta = 0$ (the variance V_n is bounded) and $\eta = \gamma$. We then recover the basic results of classical SA. On the other hand, if $b < \infty$, then we must take $p > 0$, otherwise we get $-\eta \geq 0$. Of course, the asymptotically optimal growth rate for $E[T_n]$ is not always of the form $\Theta(n^p)$; for instance, for Example 1.3,

$\Theta(\log n)$ is better (see section 5). Some technical details related to those different rates are relegated to Appendix B. Finally, further discussions and concluding remarks are made in section 6.

2. Model assumptions and almost-sure convergence. For notational simplicity, we assume $\theta^* = 0$ henceforth. This can be done without loss of generality, since we can always shift (translate) the origin to make $\theta^* = 0$. We make the following assumptions.

(A0) One has $a_n = n^{-\gamma}$, where $1/2 < \gamma \leq 1$. Also, $E\|\theta_1\|^2 < \infty$.

(A1) There are constants $\beta > 0$ and $K_\beta \geq 0$ such that $\gamma + \beta > 1$ and for all n ,

$$\|B_n\| \leq K_\beta n^{-\beta} \text{ w.p.1 and } E\|B_n\|^2 \leq K_\beta^2 n^{-2\beta}.$$

(A2) There are constants $\delta \in \mathbb{R}$ and $K_\delta \geq 0$ such that $2\gamma + \delta > 1$ and for all n ,

$$V_n = E\|\psi_n - E_n\psi_n\|^2 \leq K_\delta n^{-\delta}.$$

Condition (A0) gives an explicit form for a_n . If in lieu of $n^{-\gamma}$, we use $\Gamma_0 n^{-\gamma}$, where Γ_0 is a constant or a positive-definite matrix, then all the results of this paper carry over with minor modifications. However, for notational simplicity, we choose the current setting. Introducing Γ_0 is in fact equivalent to rescaling $\alpha_\theta(\cdot)$ and ψ_n by left multiplying them by Γ_0 and then using $a_n = n^{-\gamma}$. Therefore, there is no loss of generality in assuming $\Gamma_0 = I$, and the algorithm is thus much simpler. On the other hand, doing such a rescaling generally has an important impact on the convergence speed of the SA algorithm (1.1) and finding the “optimal” rescaling (or even a good one) is generally hard. SA algorithms with averaging (see section 6) address that problem.

In most applications, especially in Monte Carlo optimization via simulation, the initial value θ_1 is usually a deterministic constant, and the second part of (A0) holds trivially. It is also typically verified when θ_1 is random.

Condition (A1) requires that the bias decreases as $O(n^{-\beta})$. The bound holds uniformly in ω (the sample point); i.e., K_β does not depend on ω (or is nonrandom). To check that condition, we should normally look for a bound on the conditional bias as a function of the computational effort (e.g., number of replications) uniform over (s_n, θ_n) . Condition (A2) would be verified in a similar way. See [29, 30] for examples of how that can be achieved. In (A2), δ can be either positive, zero, or negative, which corresponds to a variance that (asymptotically) decreases, remains bounded, or increases with n , respectively. The condition that $2\gamma + \delta > 1$ forces the variance not to increase too fast. $K_\beta = 0$ would mean that there is no bias; $K_\delta = 0$ stands for zero variance.

To carry out the convergence analysis, we also need the following additional assumptions on the function $\alpha_\theta(\cdot)$.

(A3) The function (gradient) $\alpha_\theta(\cdot)$ is continuous.

(A4) The ODE (ordinary differential equation)

$$(2.1) \quad \dot{\theta} = -\alpha_\theta(\theta)$$

has a unique solution for each initial condition and $\alpha_\theta(\theta) = 0$ has a unique root ($= \theta^*$).

(A5) There is a finite constant $K_\alpha > 0$ such that $\|\alpha_\theta(\theta)\| \leq K_\alpha(1 + \|\theta\|)$ and there is an $\varepsilon > 0$ and a matrix H such that $\alpha_\theta(\theta) = H\theta + O(\|\theta\|^2)$ for $\|\theta\| \leq \varepsilon$. All eigenvalues of H have positive real parts. Let $\lambda_{\min} > 0$ be the minimum of those real parts.

Remark 2.1. Assumption (A4) requires the corresponding ODE to be “well behaved.” Assumption (A5) requires the function α to be quadratic near the optimizer. In the one-dimensional case, this means that the second derivative must be bounded away from zero. It also requires α_θ to grow at most linearly. To derive our convergence rate results in Theorem 3.1 and section 4, when $\gamma = 1$, we will need $\lambda_{\min}$ to be “large enough.” If it is not, then the problem should be rescaled by some matrix Γ_0 , as discussed previously. In principle, one can always make $\lambda_{\min}$ as large as needed by rescaling with an appropriate Γ_0 . On the other hand, finding the appropriate Γ_0 is not necessarily easy. If the growth rate of the objective function is not linearly bounded, then one possible solution is to project θ_n over a compact set G at each step of (1.1), as in Example 1.1 (see also section 6 or Komlós and Révész [18], Kushner and Clark [20], Azadivar and Talavage [2], and the references therein). That compact set could vary adaptively between iterations, yielding a SA algorithm with varying bounds, as suggested in, for example, Chen and Zhu [4], Yin and Zhu [45], and Andradóttir [1].

Before studying convergence rates, we first state the convergence of the algorithm in the sense of w.p.1. By virtue of the ODE approach developed in Kushner and Clark [20] (see in particular the argument of their Theorem 2.3.1), the following proposition holds.

PROPOSITION 2.1. *Under assumptions (A0)–(A5), $\theta_n \rightarrow \theta^*$ w.p.1 as $n \rightarrow \infty$.*

3. An asymptotic bound on the MSE $E\|\theta_n - \theta^*\|^2$. We now give an upper bound on $E\|\theta_n - \theta^*\|^2$, again assuming that $\theta^* = 0$. As a first step, the next theorem gives such a bound as a function of n . Then, in Corollary 3.2, the bound is related to the expected computational expenditures $E[T_n]$ and $E[C_n]$, for the case of a polynomially-growing (in n) $E[C_n]$. Note that the condition (3.2) is not really restrictive; see the remarks that follow (A5). The proof of Theorem 3.1 is given in Appendix A.

THEOREM 3.1. *Under the conditions (A0)–(A5), with $\gamma < 1$, there is an n_0 such that for all $n \geq n_0$,*

$$(3.1) \quad E\|\theta_{n+1}\|^2 \leq \frac{K_\beta^2}{\lambda_{\min}^2} n^{-2\beta} + \frac{3K_\delta}{2\lambda_{\min}} n^{-\gamma-\delta} + O(n^{-2\beta-1} + n^{-\gamma-\delta-1}) \\ = O(n^{-(2\beta \wedge (\gamma+\delta))}) = O(n^{-\kappa}).$$

The estimate above also holds for $\gamma = 1$, under the additional condition that

$$(3.2) \quad \lambda_{\min} > \max\{2\beta, 1 + \delta, 1\}.$$

Remark 3.1. We can see from the previous theorem how the convergence rate is tied up with the values of β and δ . In absence of bias, the order is purely determined by how the “noise” behaves. In particular, for $\delta = 0$ (bounded variance), our result agrees with the classical result (cf. Kushner [19]) which says that $E\|\theta_n\|^2 = O(n^{-\gamma})$. If $\gamma + \delta > 2\beta$, then the error bound becomes $E\|\theta_n\|^2 = O(n^{-2\beta})$; i.e., the convergence speed depends on how fast the bias diminishes. If $\delta < 0$, then $2\beta \wedge (\gamma + \delta) < \gamma$, so the convergence rate will be slower than that of the classical algorithm no matter how fast the bias tends to 0. If $\delta = 0$, the order cannot be better than $O(n^{-\gamma})$. Finally, for the totally degenerate case where $\beta = \delta = \infty$, that is, no bias and no noise, the theorem implies that $E\|\theta_n\|^2$ converges faster (asymptotically) than $n^{-\kappa}$ for any κ , which agrees with the classical results of nonlinear programming.

We now express the MSE bound in terms of the computer budget for the important special case where $E[T_n]$ grows polynomially fast: $n^{-p}E[T_n] \rightarrow \kappa_1$ for some $p \geq 0$.

The total computing budget $C_n = \sum_{i=1}^n T_i$ then satisfies $n^{-p-1}E[C_n] \rightarrow \kappa_1/(p+1)$ as $n \rightarrow \infty$. We further assume that the bias and variance are bounded uniformly in terms of the computer budget T_n , as follows: $(E[T_n])^b \|B_n\| \leq K_b$ w.p.1 and $(E[T_n])^d V_n \leq K_d$ for some positive (fixed) constants K_b and K_d . The convergence rate in Corollary 3.2 depends on p in a significant way.

COROLLARY 3.2. *Under the above assumptions and the conditions of Theorem 3.1, with $n^{-\beta}$ and $n^{-\delta}$ in (A1) and (A2) replaced by $(E[T_n])^{-b}$ and $(E[T_n])^{-d}$, respectively, we have that*

$$(3.3) \quad E\|\theta_{n+1}\|^2 = O((E[C_n])^{-(2bp) \wedge (\gamma+dp)/(p+1)}).$$

Proof. This is a direct consequence of Theorem 3.1. Replace β , δ , K_β , and K_δ in Theorem 3.1 by bp , dp , $K_b\kappa_1^{-b}$, and $K_d\kappa_1^{-d}$, respectively, and observe that $n^{-1}(E[C_n])^{1/(p+1)} = \Theta(1)$. $\square$

4. Asymptotic distribution of $n^{\kappa/2}\theta_n$. In this section, we devote our attention to studying the asymptotic distribution of a suitably scaled sequence of the estimation errors by means of the weak convergence methods. Again, we assume that $\theta^* = 0$ for notational simplicity. In the literature, the rate of convergence is often studied through the asymptotic distribution of a normalized error sequence. The most relevant question in the present context is to derive a nontrivial limit result for $n^{\kappa/2}\theta_n$ and calculate explicitly the asymptotic covariance matrix. The exponent $\kappa/2$ represents the rate of convergence (see, e.g., Kushner and Clark [20, p. 233]). The covariance matrix of the limiting normal distribution is another important characterization of the convergence speed. Such results are more precise than the bounds obtained in the previous section and may provide further opportunities to improve the performance of the algorithms. Limit theorems of that sort already appeared in Fabian [7] and earlier papers cited there, under assumptions different than ours. The derivation of limit theorems, in the analysis that follows, exploits the roles played by the bias and variance under different sets of assumptions and indicates the dominating factor and main influence to the asymptotic distribution and rate of convergence in each case. In the next section, those results will be applied to several examples, including those introduced in section 1.

Again, we first derive the results in terms of n and later discuss their interpretation in terms of C_n . In view of (3.3), the relative values of 2β and $\gamma + \delta$ are expected to play an important role in the result. For simplicity, the discussion is focused on the case of $\gamma = 1$, which is the most common case in practice. Similar results hold for the case $1/2 < \gamma < 1$, although one then needs to work with an interpolated sequence as in Kushner and Huang [21] to obtain the limit theorems. In order not to disrupt the flow of discussion, the proofs of the technical results are placed in Appendix A. Our study will be divided into the following three cases: (1) $1 + \delta < 2\beta$, (2) $1 + \delta > 2\beta$, and (3) $1 + \delta = 2\beta$.

4.1. Case 1: $1 + \delta < 2\beta$. In this case, from Theorem 3.1, one has

$$E\|\theta_n\|^2 = O(n^{-(1+\delta)}).$$

To obtain the desired asymptotic distribution, we first need to determine an appropriate scaling. We wish to select a real number $\kappa > 0$ such that $n^{\kappa/2}\theta_n$ converges weakly to a nontrivial distribution. It turns out that in this case, the correct order is given by $\kappa = 1 + \delta$, which is motivated by the classical central limit theorem. We derive a functional invariance principle, or functional central limit theorem, which shows how

the scaled sequence evolves as a stochastic process. This gives a stronger result than convergence in distribution alone.

To begin with, define

$$W_n(t) = \frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} \theta_{\lfloor nt \rfloor + 1}$$

for $t \in [0, 1]$, where $\lfloor z \rfloor$ denotes the integral part of z , for $z \in \mathbb{R}$. Assume that the following conditions hold.

(A6) All eigenvalues of $\tilde{H} = H - (1 + \delta)I/2$ have positive real parts.

(A7) There is a positive definite matrix R such that

$$(4.1) \quad n^\delta [E_n \psi_n - \psi_n][E_n \psi_n - \psi_n]' \xrightarrow{P} R$$

as $n \rightarrow \infty$, where $\xrightarrow{P}$ denotes convergence in probability.

Remark 4.1. If Condition (A6) fails to hold for the original problem, then the problem could always be rescaled by some appropriate Γ_0 , as discussed in Remark 2.1, in such a way that (A6) holds for the rescaled problem.

THEOREM 4.1. *Suppose that the conditions (A0)–(A7) and (3.2) are in force. Then $W_n(\cdot)$ converges weakly to a process $W(\cdot)$ which has independent Gaussian increments and covariance matrix Σt , where*

$$(4.2) \quad \Sigma = \int_0^\infty \exp(-\tilde{H}u) R \exp(-\tilde{H}'u) du.$$

Setting $t = 1$ in Theorem 4.1, the following corollary is immediate, which gives us the usual asymptotic normality result although the scaling factor $n^{(1+\delta)/2}$ is different from the classical theorems due to the formulation of our problem. For $\delta = 0$, we recover the canonical convergence rate of the classical results. In the case where the parameter θ is one dimensional, (4.2) simplifies to $\Sigma = R/(2\tilde{H})$, where $\tilde{H} = \alpha_{\theta\theta}(0) - (1 + \delta)/2$.

COROLLARY 4.2. *Under the conditions of Theorem 4.1, $n^{(1+\delta)/2}\theta_n$ converges in distribution to the normal distribution $N(0, \Sigma)$.*

Theorem 4.1 and Corollary 4.2 indicate that in case $1 + \delta < 2\beta$, the bias diminishes rather fast and is asymptotically negligible compared to the variance. The next case is the opposite situation.

4.2. Case 2: $2\beta < 1 + \delta$. In this case, the bias term becomes the dominating factor. Similar to Case 1, one may wish to have a limiting distribution result for the scaled sequence $n^\beta \theta_n$. We show in what follows that the limiting distribution is degenerate. To proceed, suppose that the following assumptions hold.

(A8) All eigenvalues of $H_b = H - \beta I$ have positive real parts.

(A9) There is a $\bar{B} \in \mathbb{R}^r$ such that as $n \rightarrow \infty$,

$$n^\beta B_n \xrightarrow{P} -\bar{B},$$

where B_n is the bias defined in (1.2).

THEOREM 4.3. *Suppose that the conditions of Theorem 4.1 are satisfied, with (A6) and (A7) replaced by (A8) and (A9), respectively. Then, as $n \rightarrow \infty$,*

$$n^\beta \theta_n \xrightarrow{P} H_b^{-1} \bar{B}.$$

Remark 4.2. Owing to (A8), H_b is nonsingular. Again, if (A8) does not hold for the original problem, then one should rescale the problem as suggested in Remarks 2.1 and 4.1. Theorem 4.3 indicates that the limiting distribution of $n^\beta \theta_n$ is degenerate. Hence one cannot expect any asymptotic normality. As n gets large, the noise effect becomes negligible and $\theta_n \sim n^{-\beta} H_b^{-1} \bar{B}$ in the sense of in probability. There is also a functional form of that limiting result. However, due to the degenerate nature, the functional limit does not provide much more than the current statement of the theorem.

4.3. Case 3: $2\beta = 1 + \delta$. In accordance with the results of the last two subsections, we find that in the current situation both the noise and the bias contribute to the asymptotic distribution. The next theorem shows that asymptotic normality holds in this case. However, the limiting normal distribution is generally not centered; its mean is given by $H_b^{-1} \bar{B}$ ($= \tilde{H}^{-1} \bar{B}$ in this case), which is also the constant $n^\beta \theta_n$ converges to in Case 2 of the previous subsection.

THEOREM 4.4. *Assume that the conditions of Theorems 4.1 and 4.3 are satisfied with $2\beta = 1 + \delta$. Then*

$$n^{(1+\delta)/2} \theta_n - H_b^{-1} \bar{B} \xrightarrow{d} N(0, \Sigma)$$

as $n \rightarrow \infty$, where Σ is given by (4.2) and $\xrightarrow{d}$ denotes convergence in distribution.

Theorem 4.4 could be viewed in a sense as a generalization of the results of the two previous cases. Indeed, if (A6)–(A9) hold with $1 + \delta < 2\beta$, then (A9) will hold with $\bar{B} = 0$ and β replaced by $(1 + \delta)/2 < \beta$. Then, applying Theorem 4.4 with β replaced by $(1 + \delta)/2$, we recover Corollary 4.2. Similarly, if $1 + \delta > 2\beta$, then, with δ replaced by $2\beta - 1 < \delta$, (A7) holds with $R \equiv 0$ and we also recover Theorem 4.3 from Theorem 4.4.

4.4. Asymptotic distribution in terms of the budget C_n . The results obtained in this section can easily be transferred to statements with $n^{\kappa/2}$ replaced by $C_n^{\eta/2}$ for some real number η . To be more specific, suppose that

$$(A10) \quad n^{-p-1} C_n \rightarrow \kappa_b \text{ in probability for some } \kappa_b > 0.$$

Under the conditions of the previous theorems, we obtain the following.

COROLLARY 4.5. *Assume that conditions (A0)–(A10) and (3.2) are satisfied. If $1 + \delta < 2\beta$, then*

$$(C_n / \kappa_b)^{(1+\delta)/(2(p+1))} \theta_n \xrightarrow{d} N(0, \Sigma) \quad \text{as } n \rightarrow \infty.$$

If $1 + \delta > 2\beta$, then

$$(C_n / \kappa_b)^{\beta/(p+1)} \theta_n \xrightarrow{p} H_b^{-1} \bar{B} \quad \text{as } n \rightarrow \infty.$$

If $1 + \delta = 2\beta$, then

$$(C_n / \kappa_b)^{\beta/(p+1)} \theta_n - H_b^{-1} \bar{B} \xrightarrow{d} N(0, \Sigma) \quad \text{as } n \rightarrow \infty.$$

Remark 4.3. If $n^{-p} T_n \rightarrow \kappa_1$ in probability as $n \rightarrow \infty$, then (A10) holds with $\kappa_b = \kappa_1 / (p+1)$. In accordance with Theorem 2 of Glynn and Whitt [16], if $n^{-p} E_n[T_n] \rightarrow \kappa_1$ w.p.1 and $n^{-2p-1+\varepsilon} E[T_n - E_n[T_n]]^2 \rightarrow \kappa_2$ w.p.1 for some positive constants κ_1, κ_2 , and ε , then $n^{-p-1} C_n \rightarrow \kappa_b = \kappa_1 / (p+1)$ w.p.1. Hence (A10) holds with $\kappa_b = \kappa_1 / (p+1)$. Note, however, that only convergence in probability is needed in our result.

5. Specific setups and examples. We now study the implications of our results in different situations related to the examples given in the introduction. We shall assume in this section that $\gamma = 1$ and (sometimes implicitly) that the functions considered satisfy (A3)–(A10) and (3.2). We analyze a series of rather general cases arising in the simulation of stochastic discrete-event systems, most of which are summarized in Tables 5.1–5.3. In view of our previous results, we say that the *optimal asymptotic rate* of convergence is reached if, with the largest possible η , either (1.5) holds or $C_n^{\eta/2}\theta_n$ converges in distribution (with a possibly degenerate limit). We spell out the conditions needed, but relegate some technical details to Appendix B. It turns out that for several situations of interest, the optimal asymptotic rate can be reached for any (nonnegative) polynomial rate of increase of T_n ; that is, either by spending increasingly more time to get better estimates or by going quickly with cheap estimates.

From Corollary 4.5, one can obtain not only the convergence rate but also the asymptotic MSE of $C_n^{\eta/2}\theta_n$ as a function of the other asymptotic constants in the problem. We give examples of that in Appendix B. In principle, one could then minimize the asymptotic MSE as well. This may be hard to implement, because the asymptotic constants such as $\bar{B}$, H_b , etc., are typically unknown, but is nevertheless interesting to study from the theoretical point of view.

5.1. Finite-horizon models.

Example 5.1. Let $\alpha(\theta) = E[X]$, where X is a random variable whose distribution $F(\theta, \cdot)$ depends on θ , and for which i.i.d. samples can be obtained via simulation (or by any other method). We suppose that the function α satisfies Assumptions (A3)–(A5), and we want to minimize this function with respect to θ , via SA. At each iteration of SA, we need a gradient estimator ψ_n at $\theta = \theta_n$. A classical way of estimating the gradient $\alpha_\theta(\theta)$ is through the use of FDs (see, e.g., Kushner and Clark [20], L’Ecuyer and Perron [27], or Zazanis and Suri [46] for more details on FD methods). To simplify the notation here, let θ be a scalar. A generalization to the multidimensional case is straightforward and the rates (e.g., in Table 5.1) are the same (see also Fabian [7] for multidimensional results).

One FD estimator of $\alpha_\theta(\theta)$ is given by

$$(5.1) \quad \psi = \frac{X^+ - X^-}{2c},$$

where X^+ and X^- are independent random variables (r.v.s) with distributions $F(\theta^+, \cdot)$ and $F(\theta^-, \cdot)$, respectively, $\theta^+ = \theta + c$, $\theta^- = \theta - c$, and $c \geq 0$ is the half size of the FD interval. The latter estimator is called a *central* (or two-sided) FD estimator with independent random numbers and we shall refer to it by the acronym FDC. A forward (or one-sided) estimator, which we call FDF, is defined by

$$(5.2) \quad \psi = \frac{X^+ - X}{c},$$

where X now has distribution $F(\theta, \cdot)$ and is also independent of X^+ . Both estimators are biased. Their bias decreases to 0 as $c \rightarrow 0$, but then their variance increases towards infinity.

To reduce the variance, we can take the average of several copies of ψ at any given iteration of the SA algorithm. Suppose that at step n of SA, we generate N_n i.i.d. replicates of ψ as defined by either (5.1) or (5.2), with $\theta = \theta_n$ and $c = c_n$, and

take the average as a derivative estimator ψ_n . Let $\psi_{n,i}$ denote the i th copy of ψ thus generated at step n and $\tau_{n,i}$ be the (random) computing time required to generate it. The pairs $(\psi_{n,i}, \tau_{n,i})$, $1 \leq i \leq N_n$, are then i.i.d., the gradient estimator at step n is

$$\psi_n = \frac{1}{N_n} \sum_{i=1}^{N_n} \psi_{n,i},$$

and the computational expenditure at step n is $T_n = \sum_{i=1}^{N_n} \tau_{n,i}$ (we neglect the time for computing the average and for updating θ_n from ψ_n in (1.1)). Here, N_n is deterministic, but T_n and $C_n = \sum_{i=1}^n C_i$ (the cumulated computing time up to step n) are random.

Suppose that the sequences $\{N_n\}$ and $\{c_n\}$ are chosen such that (as $n \rightarrow \infty$) $n^{-p}N_n \rightarrow \kappa_N$ and $n^\nu c_n \rightarrow \kappa_c$ for some constants $p \geq 0$, $\nu > 0$, $\kappa_N > 0$, and $\kappa_c > 0$. Thus, ν reflects the rate of decreasing of the finite-difference interval (as a function of the iteration number), while p is the rate of increase of the number of replicates of ψ (or computing budget) per SA iteration. We make the (reasonable) assumption that

$$(5.3) \quad \sum_{i=1}^n T_i / \sum_{i=1}^n N_i \xrightarrow{\text{w.p.1}} \kappa_T$$

as $n \rightarrow \infty$, where κ_T represents the average cost for computing ψ . The left side in (5.3) is the average computing time per replicate of ψ computed during the first n iterations of SA. If θ was fixed, then (5.3) would follow directly from the strong law of large numbers, so this assumption is quite reasonable when θ converges to the optimizer.

Under these conditions, we have that as $n \rightarrow \infty$, w.p.1,

$$C_n = \sum_{i=1}^n T_i \sim \kappa_T \sum_{i=1}^n N_i \sim \kappa_T \kappa_N \sum_{i=1}^n n^p \sim \kappa_T \kappa_N n^{p+1} / (p+1).$$

In other words, the cumulative computing cost C_n increases with n as follows:

$$n^{-p-1} C_n \xrightarrow{\text{w.p.1}} \kappa_T \kappa_N / (p+1) \stackrel{\text{def}}{=} \kappa_b.$$

(A11) Assume that (3.2) holds, that $\text{Var}[X]$ is bounded over G and converges to a positive constant as $\theta \rightarrow 0$ (recall that X has distribution $F(\theta, \cdot)$). Suppose also that α is twice continuously differentiable in G if we use FDf and three times continuously differentiable in G if we use FDC.

In Appendix B, we obtain the optimal values of p and ν , and the corresponding values of κ and η , under Assumption (A11). The optimal values are those that maximize η ; they are given in the first two lines of Table 5.1. The optimal η can be obtained by taking any $p \geq 0$, provided that ν (the convergence rate of c_n) is chosen as specified in the table. Recall that η tells us the convergence rate of θ_n in terms of the computing budget C_n , whereas κ gives the convergence rate as a function of the number of iterations. The values of β and δ in the table are the largest values that satisfy Assumptions (A1) and (A2) (or the values that satisfy (A7) and (A9)). They give the convergence rate of the “bias” $\|B_n\|$ and “variance” V_n of the gradient estimator ψ_n as functions of n . A negative value of δ means that V_n increases with n .

The case $p = 0$ means that the number N_n of replicates of ψ is the same at all iterations n . In this case, the optimal values are $(\nu, \kappa, \eta) = (1/4, 1/2, 1/2)$ for FDF

TABLE 5.1
Convergence rates of SA for a finite-horizon model ($\gamma = 1$).

Model	p	ν	β	δ	κ	η
FDf	≥ 0	$(p+1)/4$	ν	$p-2\nu$	$(p+1)/2$	$1/2$
FDc	≥ 0	$(p+1)/6$	2ν	$p-2\nu$	$2(p+1)/3$	$2/3$
FDf (CRN1)	≥ 0	$(p+1)/3$	ν	$p-\nu$	$2(p+1)/3$	$2/3$
FDc (CRN1)	≥ 0	$(p+1)/5$	2ν	$p-\nu$	$4(p+1)/5$	$4/5$
FDf (CRN2)	≥ 0	$\geq (p+1)/2$	ν	p	$p+1$	1
FDc (CRN2)	≥ 0	$\geq (p+1)/4$	2ν	p	$p+1$	1
IPA, LR	≥ 0		∞	p	$p+1$	1

and $(\nu, \kappa, \eta) = (1/6, 2/3, 2/3)$ for FDc. These values are well known (see, e.g., Fabian [7], Kushner and Clark [20], and Kushner and Huang [21]), and the SA algorithm is then called the *Kiefer–Wolfowitz* algorithm.

For $p > 0$, N_n increases with n , so we spend more and more time per iteration as n goes up. This increases the cost per iteration but reduces the variance of ψ_n . We can then decrease c_n at a faster rate, to make the bias decrease faster, so that the variance and squared bias decrease at the same rate. This gives the optimal ν . When $p > 0$, the convergence rate in terms of the number of iterations is “faster” than that in terms of C_n . For large enough p we even obtain a supercanonical rate ($\kappa > 1$) in terms of n (but not in terms of C_n). The optimal η is the same for all $p \geq 0$; i.e., for an increase of N_n at any polynomial rate, we get the same asymptotic efficiency, provided that c_n decreases at rate $n^{-\nu}$ with the ν specified in the table. This holds for all the “models” considered in Table 5.1. In Appendix B, we also examine the asymptotic mean and variance of $C_n^{\eta/2}\theta_n$ in terms of the asymptotic constants κ_c, κ_N , and so on.

Finite-difference estimators can be improved by using common random numbers (CRN), as explained by, e.g., Glynn [11], Glasserman and Yao [10], and L’Ecuyer and Perron [27]. The basic idea is to view X as a function of θ and ω , say, $X(\theta, \omega)$, where ω represents an underlying sample point whose distribution does not depend on θ . For example, in a simulation model, ω may represent a sequence of i.i.d. $U(0, 1)$ r.v.s used to drive the simulation. CRN means using the same ω in the same way for both X^- and X^+ in (5.1) or for both X and X^+ in (5.2). The aim is to induce a strong positive correlation between X^- and X^+ (or X and X^+) without changing their expectations. For a fixed FD interval, the bias of the FD estimator ψ is unchanged by using CRNs, but the variance can be reduced dramatically. L’Ecuyer and Perron [27] show that under Assumption (A12), the variance of ψ with CRN is $O(1)$ (i.e., bounded) as $c_n \rightarrow 0$. Under (A13), the bound is uniform over G , so (A2) holds with $\delta = p$. Recall that with independent random numbers, the variance of ψ is $O(c_n^{-2})$.

(A12) Assume that in some neighborhood of 0, w.p.1, $X(\theta) \equiv X(\theta, \omega)$ is a continuous function of θ , differentiable except perhaps at a denumerable number of points. Also, the derivative $X_\theta(\theta)$, where it exists, is uniformly bounded by a square-integrable r.v. (independent of θ).

(A13) Assumption (A12) holds with the neighborhood of 0 replaced by G .

We shall denote the CRN approach by CRN2 when (A13) is satisfied. The optimal asymptotic rate of $\eta = 1$ is obtained under the conditions specified in Table 5.1: Take any $p \geq 0$, $\nu \geq (p+1)/2$ for FDf, and $\nu \geq (p+1)/4$ for FDc. So, it suffices that $c_n \rightarrow 0$ fast enough. In fact, one can as well take “ $c_n = 0$ ” for all n , i.e., take the limit

as $c \rightarrow 0$ in (5.1) or (5.2), which yields the sample derivative $X_\theta(\theta)$. Under (A13), this sample derivative turns out to be an unbiased derivative estimator with variance uniformly bounded over G . It is called the *infinitesimal perturbation analysis* (IPA) estimator (Glasserman [9]). For CRN2 and IPA, θ_n converges at rate $O(C_n^{-1/2})$, which is the canonical rate in terms of C_n . The convergence in terms of n is supercanonical ($\kappa > 1$) when $p > 0$.

If (A13) fails to hold, the variance may still increase much slower, as $c \rightarrow 0$, with CRN than with independent random numbers. Under the following conditions, for instance, the variance of ψ is improved to $O(c^{-1})$ (not quite as good as $O(1)$, but better than $O(c^{-2})$).

(A14) There exist finite constants K_1 , K_2 , and $\bar{c}$, such that

$$\sup_{\theta_0 \in G} \sup_{\|\theta - \theta_0\| \leq \bar{c}} |X(\theta, \omega) - X(\theta_0, \omega)| \leq K_1 \quad \text{w.p.1}$$

and for all $c \leq \bar{c}$

$$P[X(\cdot, \omega) \text{ is continuous in } (\theta_0 - c, \theta_0 + c)] \geq 1 - K_2 c.$$

Using an argument similar to that in the proof of Proposition 3 of L'Ecuyer and Perron [27], one can show that $\text{Var}[\psi] = O(c^{-1})$ (uniformly in θ) under (A14). We may further assume (as in Glynn [11]) that there is a constant $\sigma_D > 0$ such that $c \text{Var}[\psi] \rightarrow \sigma_D^2$ as $c \rightarrow 0$ and $\theta \rightarrow 0$, and call this the CRN1 setup. The optimal asymptotic rates for CRN1 are $\eta = 2/3$ for FDF and $\eta = 4/5$ for FDC, which is a significant improvement over the case of independent random numbers. These rates are obtained under the conditions given in Table 5.1: $\nu = (p+1)/3$ for FDF and $\nu = (p+1)/5$ for FDC.

Besides IPA, there is another approach for obtaining an unbiased derivative estimator, called the LR method (Glynn [12], L'Ecuyer [25], and Rubinstein and Shapiro [35]). Its variance decreases at the same rate as that of IPA as a function of the computing budget, but it is typically larger by a constant factor. On the other hand, there are several classes of systems for which the LR method applies whereas IPA (in its direct form) does not.

Example 5.2. Let us return to Example 1.2. Suppose that g is twice continuously differentiable, that all X_j and Y_j have finite moments of all orders, and that (A3)–(A5) hold. The gradient can be written as

$$\begin{aligned} \alpha_\theta(\theta) &= \sum_{j=1}^d g_j(\mu_1(\theta), \dots, \mu_d(\theta)) \mu_{j,\theta}(\theta) \\ &\stackrel{\text{def}}{=} \tilde{g}(\mu_1(\theta), \dots, \mu_d(\theta), \mu_{1,\theta}(\theta), \dots, \mu_{d,\theta}(\theta)). \end{aligned}$$

Suppose that the gradient estimator ψ_n at step n is ψ , as given in Example 1.2, based on $N = N_n$ replications of X_j and Y_j , whose bias and variance are assumed to be bounded uniformly in θ , and let p , κ_N , T_n , κ_T , and κ_b be as in Example 5.1. For $1 \leq i, k \leq 2d$, let Γ_{ik} denote the (i, k) th entry of the covariance matrix $\Gamma = \text{Cov}(X_1, \dots, X_d, Y_1, \dots, Y_d)$, let $\tilde{g}_i$ be the i th component of the gradient of $\tilde{g}$, and let $\tilde{g}_{ik}$ be the (i, k) th entry of the Hessian of $\tilde{g}$. One has

$$\tilde{g}_i = \begin{cases} \sum_{j=1}^d g_{ij}(\mu) \mu_{j,\theta}(\theta) & \text{for } i = 1, \dots, d, \\ g_i(\mu) & \text{for } i = d+1, \dots, 2d, \end{cases}$$

and

$$\tilde{g}_{ik} = \begin{cases} \sum_{j=1}^d g_{ijk}(\mu) \mu_{j,\theta}(\theta) & \text{for } 1 \leq i, k \leq d, \\ g_{ik}(\mu) & \text{for } 1 \leq i \leq d < k \leq 2d, \\ 0 & \text{for } d < i, k \leq 2d, \end{cases}$$

where $\mu = (\mu_1(\theta), \dots, \mu_d(\theta))$, while g_{ik} and g_{ijk} denote the second and third derivatives of g with respect to its parameters (i, k) and (i, j, k) , respectively. By applying the results of Glynn and Heidelberger [13] and Glynn and Whitt [16, Example 2] to $\tilde{g}$, we find that (A7) and (A9) hold with $\beta = \delta = p$, $\bar{B} = -(1/2) \sum_{i=1}^d \sum_{k=1}^d \tilde{g}_{ik} \Gamma_{ik}$, and $R = \sum_{i=1}^d \sum_{k=1}^d \tilde{g}_i \tilde{g}_k \Gamma_{ik}$.

It then follows from Corollary 4.5 that

$$(5.4) \quad C_n^{1/2} \theta_n \Rightarrow \begin{cases} \kappa_b^{p/(p+1)} H_b^{-1} \bar{B} & \text{if } 0 \leq p < 1, \\ \kappa_b^{1/2} N(H_b^{-1} \bar{B}, \Sigma) & \text{if } p = 1, \\ \kappa_b^{1/2} N(0, \Sigma) & \text{if } p > 1. \end{cases}$$

This gives $\eta = 1$ for $p \geq 1$. To get the best asymptotic properties, one should choose $p > 1$ because the asymptotic bias vanishes while the asymptotic covariance matrix remains the same as for $p = 1$.

As a special case, suppose that we are interested in minimizing a ratio of expectations, so $g(\mu_1(\theta), \mu_2(\theta)) = \mu_1(\theta)/\mu_2(\theta)$ and

$$\tilde{g}(\mu_1(\theta), \mu_2(\theta), \mu_{1,\theta}(\theta), \mu_{2,\theta}(\theta)) = \frac{\mu_2(\theta) \mu_{1,\theta}(\theta) - \mu_1(\theta) \mu_{2,\theta}(\theta)}{\mu_2^2(\theta)}.$$

The components of the gradient of $\tilde{g}$ are then

$$\begin{aligned} \tilde{g}_1 &= -\mu_{2,\theta}(\theta)/\mu_2^2(\theta), \\ \tilde{g}_2 &= \mu_{1,\theta}(\theta)/\mu_2^2(\theta), \\ \tilde{g}_3 &= 1/\mu_2^2(\theta), \\ \tilde{g}_4 &= -\mu_1(\theta)/\mu_2^2(\theta), \end{aligned}$$

and the second-order derivatives $\tilde{g}_{ik}$ are easily obtained by differentiating these expressions. The matrix Γ is highly problem dependent.

5.2. Steady-state models.

Example 5.3. Suppose we want to minimize the infinite-horizon time average of a real-valued stochastic process Z , whose evolution depends on θ :

$$\alpha(\theta) \stackrel{\text{def}}{=} \lim_{t \rightarrow \infty} \frac{1}{t} E \int_0^t Z(\theta, s) ds.$$

At step n of the SA algorithm, we intend to estimate the gradient $\alpha_\theta(\theta_n)$ by its truncated version $\alpha_\theta(\theta_n, t_n)$, defined as the gradient of

$$\alpha(\theta_n, t_n) \stackrel{\text{def}}{=} \frac{1}{t_n} E \int_0^{t_n} Z(\theta, s) ds.$$

For (fixed) finite t_n , the latter can be estimated by any of the methods discussed in Example 5.1. However, the bias and variance will also depend on the truncation point t_n . Let $\{t_n, n \geq 1\}$ be a deterministic sequence such that $n^{-q} t_n \rightarrow \kappa_t > 0$ as $n \rightarrow \infty$ for some constants q and κ_t . We shall assume the following.

TABLE 5.2
Convergence rates for the truncated-horizon steady-state model ($\gamma = 1, p \geq 0$).

Model	q	ν	β	δ	κ	η
FDf	$\geq (p+1)/3$	$(p+q+1)/4$	ν	$2\nu - 1$	2ν	$1/2$
FDc	$\geq (p+1)/2$	$(p+q+1)/6$	2ν	$4\nu - 1$	4ν	$2/3$
FDf (CRN1)	$\geq (p+1)/2$	$(p+q+1)/3$	ν	$2\nu - 1$	2ν	$2/3$
FDc (CRN1)	$\geq 2(p+1)/3$	$(p+q+1)/5$	2ν	$4\nu - 1$	4ν	$4/5$
FDf (CRN2)	$\geq p+1$	$\geq (p+q+1)/2$	$\nu \wedge q$	$p+q$	$p+q+1$	1
FDc (CRN2)	$\geq p+1$	$\geq (p+q+1)/4$	$2\nu \wedge q$	$p+q$	$p+q+1$	1
IPA	$\geq p+1$		q	$p+q$	$p+q+1$	1
LR	$(p+1)/3$		q	$p-q$	$2q$	$1/2$
CLR	$(p+1)/2$		q	p	$2q$	$2/3$

(A15) As $t \rightarrow \infty$, the bias component due to the truncated horizon satisfies $t(\alpha_\theta(\theta, t) - \alpha_\theta(\theta)) \rightarrow \kappa_1(\theta)$, where $\sup_\theta |\kappa_1(\theta)| < \infty$, and the variance of the estimator of $\alpha(\theta, t)$ (for the FD methods) or of $\alpha_\theta(\theta, t)$ (for IPA) decreases as $O(t^{-1})$ uniformly in θ .

Assumption (A15) typically holds and can be proved rigorously for a large class of regenerative models (L'Ecuyer and Glynn [30]). L'Ecuyer [26] studies the convergence rates of several gradient estimators in that context; here, we are rather interested in the convergence rate of SA. Notice that even when the model is regenerative, it is often much more convenient to use truncated-horizon estimators instead of estimators directly based on the regenerative cycles, because for the former there is no need to recognize the regeneration epochs during the simulation. These epochs may be rare, or their identification may require nonnegligible effort, especially if the system is modeled as a Harris-recurrent Markov chain (see, e.g., Sigman and Wolff [39] or Meyn and Tweedie [31]). Table 5.2 tells us how the values of p , q , and ν can be chosen in order to maximize η for the different types of gradient estimators. See Appendix B for the technical details. A typical choice would be to take a single replication at each SA iteration ($p = 0$) and use a horizon length $t_n = \lfloor \kappa_t n^q \rfloor$ at iteration n , where q is defined by its lower bound in the table. For example, for FDc under CRN1, take $q = 2/3$ and $\nu = 1/3$, so $t_n = \Theta(n^{2/3})$, $c_n = \Theta(n^{-1/3})$, $\|B_n\| = O(n^{-2/3})$, $V_n = O(n^{1/3})$, and $E[\|\theta_n\|^2] = O(n^{-4/3}) = O(C_n^{-4/5})$. The constraints given in Table 5.2 only make sure that η is maximized; a more refined analysis would be required to also minimize the asymptotic MSE constant K_{MSE} defined by (B.1) in Appendix B. Again, we see that $E\|\theta_n\|^2$ may converge faster than $O(n^{-1})$ as a function of n (for large values of $p+q$), but not faster than $\Theta(C_n^{-1})$ as a function of the total budget. The optimal values of p and ν here are not the same as for the finite-horizon case (Table 5.1); however, the same η is reached. For example, for FDc with independent random numbers, $p = 0$ and $\nu = 1/6$ give the optimal rate in the finite-horizon case, but not in the infinite-horizon case.

In the infinite-horizon case, the LR estimator typically behaves much differently than IPA since its variance typically increases linearly w.r.t. the horizon length t_n (Glynn [12] and Rubinstein and Shapiro [35]). One must then select a small value of q to control the variance, and as a result the overall convergence turns out to be quite slow ($\eta = 1/2$). There is also a *control-variate* variant of LR (L'Ecuyer [26]) for which (under some conditions) the variance V_n is $\Theta(1)$. It is denoted by CLR in the table. $\square$

Example 5.4. Let us return to the $GI/GI/1$ queue of Example 1.1. For that problem, Glynn and L'Ecuyer [14] have proven the convergence w.p.1 of SA (using projection over a finite interval) combined with each of the gradient estimation methods discussed so far in this section under a given set of assumptions. They did not study the convergence rates. By combining their methods of proofs with the results of L'Ecuyer and Glynn [30], it is easy to show that under their assumptions (A–C), together with additional assumptions on $C(\theta)$ to ensure (A3)–(A5), the exact orders found in the previous example for the truncated-horizon case apply to this model (for the common random numbers, the variant that applies is CRN2). In particular, Table 5.2 gives the appropriate convergence rates for several of the variants which were experimented with by L'Ecuyer, Giroux, and Glynn [28] for the $M/M/1$ queue and therefore explains many of the numerical results obtained by these authors.

Since the single queue considered in this example is a regenerative system, the function $\alpha(\theta)$ can also be written as a ratio of two expectations, and the methodology of Example 5.2 can be applied with g defined as $g(\mu_1, \mu_2) = \mu_1/\mu_2$. This gives $\eta = 1$ provided that $p \geq 1$.

Example 5.5. Haurie, L'Ecuyer, and Van Delft [17] proved the convergence w.p.1 of SA combined with IPA for a class of piecewise-deterministic control systems encountered in manufacturing. They did not find the convergence rates. However, under their assumptions 2.1 and 2.2, and using again the results of L'Ecuyer and Glynn [30], one can show that $\|B_n\|$ and V_n are both $\Theta(t_n^{-1})$ for the IPA estimator, so the results of Table 5.2 for IPA apply to this model, provided that (A3)–(A5) also hold for the performance measure considered.

5.3. An infinite-horizon model with discounting.

Example 5.6. We now return to the infinite-horizon discounting model of Example 1.3, assuming that $Z(\cdot)$ is strictly positive and bounded. Consider the truncated-horizon cost estimator

$$X(\theta, t) = \int_0^t e^{-\rho s} Z(s) ds,$$

and let $\alpha(\theta, t) = E[X(\theta, t)]$.

(A16) Suppose that $\text{Var}[X(\theta, t)]$ and $e^{\rho t}(\alpha(\theta) - \alpha(\theta, t))$ are both bounded away from 0 and from infinity, uniformly in θ . Let $\text{Var}[X(\theta, t)] \rightarrow \sigma^2$ and $e^{\rho t}(\alpha(\theta) - \alpha(\theta, t)) \rightarrow \kappa_\beta$ when $\theta \rightarrow 0$ and $t \rightarrow \infty$, where σ and κ_β are positive constants.

This assumption is reasonable; see Fox and Glynn [8] for justifications. Suppose also that the time required for computing $X(\theta, t)$ is approximately $\kappa_T t$ for some constant κ_T . Fox and Glynn [8] have shown that to optimize the convergence rate of the MSE when i.i.d. replicates of $X(\theta, t)$ are used to estimate $\alpha(\theta)$ under the constraint of a limited budget of size C , the horizon length t should increase in such a way that $t/\ln C \rightarrow 1/(2\rho)$ as $C \rightarrow \infty$. The budget C is then split into approximately $2\rho C/(\kappa_T \ln C)$ simulation runs of length $t = \kappa_T \ln C/(2\rho)$.

We show in Appendix B that in the SA context, if an horizon length t_n is used at iteration n , then $t_n/\ln n$ must converge to a positive constant in order to optimize the convergence rate of the MSE of θ_n , and the convergence rate actually depends on the product of that constant with the discount rate ρ . We consider IPA and the FD variants introduced in Example 5.1. In the case of IPA, we assume that

$$X_\theta(\theta, t) = \int_0^t e^{-\rho s} \tilde{Z}(s) ds$$

TABLE 5.3
Asymptotic behavior of SA: The infinite-horizon model with discounting.

Method	$\rho\kappa_t$	ν	β	δ	κ	$\tilde{\eta}$
FDf	$\geq (p+1)/2$	$(p+1)/4$	ν	$p-\nu$	$(p+1)/2$	$1/2$
FDC	$\geq (p+1)/3$	$(p+1)/6$	2ν	$p-2\nu$	$2(p+1)/3$	$2/3$
FDf (CRN1)	$\geq 2(p+1)/3$	$(p+1)/3$	ν	$p-\nu$	$2(p+1)/3$	$2/3$
FDC (CRN1)	$\geq 2(p+1)/5$	$(p+1)/5$	2ν	$p-2\nu$	$4(p+1)/5$	$4/5$
FDf (CRN2)	$\geq p+1$	$\geq (p+1)/2$	$\rho\kappa_t \wedge \nu$	p	$p+1$	1
FDC (CRN2)	$\geq (p+1)/2$	$\geq (p+1)/4$	$\rho\kappa_t \wedge 2\nu$	p	$p+1$	1
IPA	$\geq (p+1)/2$		$\rho\kappa_t$	p	$p+1$	1

is an unbiased estimator of $\alpha_\theta(\theta, t)$ and that the assumptions (A16) hold with X and α replaced by X_θ and α_θ . For the six variants of FD, $X(\theta, t)$ replaces $X(\theta)$ in the setups of Example 5.1. At step n of SA, we compute N_n i.i.d. replications of the gradient estimator ψ over a horizon-length t_n (e.g., $\psi = X_\theta(\theta_n, t_n)$ in the case of IPA, ψ given by (5.1) for FDC, etc.). The average of those N_n replications is the gradient estimator ψ_n . We suppose that $n^p N_n \rightarrow \kappa_N$ and $t_n/\ln n \rightarrow \kappa_t$ as $n \rightarrow \infty$, where $p \geq 0$, $\kappa_N > 0$, and $\kappa_t \in [0, \infty]$ are fixed constants (Here, $\kappa_t = \infty$ is a loose notation to indicate the case where t_n increases faster than $\Theta(\ln n)$.)

Table 5.3 gives the values of κ and $\tilde{\eta}$ such that $E\|\theta_n\|^2 = \Theta(n^{-\kappa}) = \Theta(((E[C_n])^{-1} \ln E[C_n])^{\tilde{\eta}})$ for the largest possible $\tilde{\eta}$ for the different cases. Note that $\tilde{\eta}$ has a slightly different meaning than η in the previous examples. Here, $(E[C_n])^{-1} \ln E[C_n] = \Theta(n^{-p-1})$, so $\tilde{\eta} = \kappa/(p+1)$. The optimal rates are attained when the constant κ_t is finite and large enough, as indicated in the second column of Table 5.3. So, the horizon-length t_n must increase at a logarithmic rate, and the factor of increase must be large enough. The optimal strategy is to take $\rho\kappa_t$ equal to (or slightly larger than) the lower bound given in the table (e.g., $\kappa_t \geq (p+1)/(2\rho)$ for IPA). The values of κ and $\tilde{\eta}$ here are the same as κ and η in Example 5.1; however, the convergence in terms of C_n is at a slightly slower rate due to the additional logarithmic factor.

6. Further remarks and conclusion. In this section, we briefly discuss other possible variants of our setup, such as SA with averaging and projection methods, then give a conclusion.

6.1. Algorithms with averaging. Very recently, some new methods were proposed and suggested for stochastic approximation by Polyak [33], Ruppert [37], and Bather [3]. See also Yin [42], Polyak and Juditsky [34], and Kushner and Yang [23]. It has been a long-time effort (dated back to Chung [5]) to improve the performance of SA algorithms. Among the choices of step size $a_n = 1/n^\gamma$ for $1/2 < \gamma < 1$, $a_n = 1/n$ gives the highest order of convergence. However, if the step size is too small in the initial stage, the iterates wonder around and take a long time to settle down. From this point of view, one may wish to choose a large step size instead. But large step size gives rise to lower rate of convergence. The averaging procedures provide a good alternative in taking care of these trade-offs. The procedures are multistep iterative schemes. With a large initial step size, they produce a “squeezing effect” forcing the iterates to reach a vicinity of $\theta^* = 0$ faster. Meanwhile, by means of averaging, they keep the convergence at the optimal rate with the smallest possible asymptotic covariance.

Inspired by the averaging approach suggested by Polyak [33] and Ruppert [37],

for our problem we may consider the algorithm

$$(6.1) \quad \theta_{n+1} = \theta_n - n^{-\gamma} \psi_n, \quad \bar{\theta}_n = \frac{1}{n} \sum_{j=1}^n \theta_j.$$

Notice that the averaging here creates no additional burden since the average can be recursively updated as $\bar{\theta}_{n+1} = \bar{\theta}_n + (\theta_{n+1} - \bar{\theta}_n)/(n+1)$.

Motivated by the work of Bather [3] (see also Schwabe [38] and Yin and Yin [44]), we may consider another algorithm, which uses averaging in both trajectories and observations (measurements). In addition to the advantages mentioned above, that algorithm appears to be more stable in the initial period (see [44]). Consider

$$(6.2) \quad \theta_{n+1} = \bar{\theta}_n - n^{-\gamma} \sum_{i=1}^n \psi_i, \quad \bar{\theta}_n = \frac{1}{n} \sum_{j=1}^n \theta_j,$$

where $1/2 < \gamma < 1$. The study of asymptotic properties of the averaging algorithms in conjunction with the setup of this paper could be carried out. The idea would be to combine the approach in Yin [42] and Yin and Yin [44] with the results of this work. This is a topic for further investigation.

6.2. Projection algorithms. The discussion in this paper is based on the basic recursive SA algorithm (1.1). Many variants of the algorithm can also be considered. For instance, our results can easily be incorporated into algorithms with projection and truncations. Such algorithms have the advantage that growth conditions on the underlying functions are no longer required.

Suppose G is a closed and convex set (for example, a closed ball or a closed hyper-rectangle in $\mathbb{R}^r$), and suppose that all the desirable equilibrium points of $\alpha(\theta)$ are interior to G . If $x \notin G$, define $\pi_G(x)$ to be the nearest point to x in G . Then the algorithm (1.1) can be replaced by

$$(6.3) \quad \theta_{n+1} = \pi_G(\theta_n - n^{-\gamma} \psi_n).$$

The limiting ODE then reads

$$\dot{\theta} = \pi_G(\alpha(\theta)).$$

The results of this paper still hold, and the discussions are the same as before. A variant of such a projection procedure is an algorithm with random truncation bounds or random projection regions along the lines of, for example, Chen and Zhu [4].

Let $q_i(\cdot)$, $i \leq \nu$, be continuously differentiable functions and

$$G = \{\theta : q_i(\theta) \leq 0 \text{ for } i = 1, 2, \dots, \nu\}.$$

Suppose that G is bounded, convex, and is the closure of its interior. If we use the projection algorithm in this case, the limit of $\{\theta_n\}$ will be related to the set of Fritz-John points or Kuhn–Tucker points (see p. 191 of Kushner and Clark [20] for more details; see also Kushner and Yin [24] for a comprehensive and updated development of constrained and unconstrained algorithms). Again, our convergence rate results apply if θ^* lies in the interior of G , but not if it is on the boundary. For more general constrained optimization problems, other methods, such as large deviations techniques (see the discussion in Chapter 10 of [24] and the references therein), may be needed to obtain the rate of convergence.

6.3. Conclusion. We have studied budget- and moment-dependent stochastic optimization algorithms and ascertained their rates of convergence for different situations arising in discrete-event simulation. The corresponding asymptotic distributions were also derived together with some far reaching functional invariance theorems. Several examples showed how those results can be used to find the “optimal” parameters of the algorithm in various SA settings. Such knowledge is important for understanding how SA algorithms behave from the theoretical point of view. As for most optimization algorithms, it is true that the practitioner rarely has all the information available (e.g., the Hessian matrix) to actually implement the algorithm with the optimal parameters and scaling. But some “rough” information could be used when available and adaptive schemes, which estimate the optimal parameters as the algorithm proceeds, may also work in some situations. Recent results suggest that SA methods with averaging are less sensitive to the choice of scaling and algorithm parameters. Studying those averaging methods within the setup of this paper is a topic of ongoing research.

Appendix A. Proofs of the theorems. This appendix provides the proofs of Theorems 3.1, 4.1, 4.3, and 4.4.

Proof of Theorem 3.1. Define $U(\theta) = (1/2)\|\theta\|^2$. We first treat the case $\gamma < 1$. The proof is divided into two steps. In Step 1, we show that $\sup_n EU(\theta_n) < \infty$. The w.p.1 convergence and the estimate above allow us to infer $EU(\theta_n) \rightarrow 0$. Then in Step 2, we refine the order of magnitude estimate.

Step 1. Since θ_n is measurable with respect to the σ -algebra that measures (n, s_n, θ_n) , one has

$$(A.1) \quad E_n[n^{-\gamma}\theta'_n(\psi_n - E_n\psi_n)] = n^{-\gamma}\theta'_n E_n[\psi_n - E_n\psi_n] = 0.$$

Take an arbitrary $0 < \varepsilon_0 \leq \varepsilon$ and an n_0 such that $2\beta n_0^{\gamma-1} \leq \varepsilon_0$. The reason for this choice will become clear in the following discussion. For the remainder of this proof, we suppose that $n \geq n_0$. Observe that

$$\|\theta'(H\theta - \alpha_\theta(\theta))\| \leq \|\theta\|(\|H\| \cdot \|\theta\| + K_\alpha(1 + \|\theta\|)) \leq K_1(1 + U(\theta))$$

for all θ and for some $K_1 > 0$. From a Taylor series expansion, using

$$(A.2) \quad \theta_{n+1} = \theta_n - n^{-\gamma}\alpha_\theta(\theta_n) + n^{-\gamma}(\alpha_\theta(\theta_n) - E_n\psi_n) + n^{-\gamma}(E_n\psi_n - \psi_n),$$

(A.1), the fact that $U_\theta(\theta_n) = \theta_n$, and (A5), we obtain that

$$(A.3) \quad \begin{aligned} EU(\theta_{n+1}) - EU(\theta_n) &= E(E_n U(\theta_{n+1}) - U(\theta_n)) \\ &= EE_n U'_\theta(\theta_n)(\theta_{n+1} - \theta_n) + (1/2)E\|\theta_{n+1} - \theta_n\|^2 \\ &= n^{-\gamma}E\theta'_n[-H\theta_n + (H\theta_n - \alpha_\theta(\theta_n)) + (\alpha_\theta(\theta_n) - E_n\psi_n)] \\ &\quad + (1/2)E\|\theta_{n+1} - \theta_n\|^2 \\ &\leq -2n^{-\gamma}\lambda_{\min}EU(\theta_n) + K_1n^{-\gamma}(1 + EU(\theta_n)) + n^{-\gamma}E\theta'_n(\alpha_\theta(\theta_n) - E_n\psi_n) \\ &\quad + (3/2)n^{-2\gamma}E(\|\alpha_\theta(\theta_n)\|^2 + \|\alpha_\theta(\theta_n) - E_n\psi_n\|^2 + \|\psi_n - E_n\psi_n\|^2). \end{aligned}$$

Using (A5), we find

$$(A.4) \quad E\|\alpha_\theta(\theta_n)\|^2 \leq 2K_\alpha^2(1 + 2EU(\theta_n)).$$

We also have

$$E\|\theta'_n(\alpha_\theta(\theta_n) - E_n\psi_n)\| \leq E^{1/2}\|\theta_n\|^2 E^{1/2}\|\alpha_\theta(\theta_n) - E_n\psi_n\|^2.$$

Using the inequality $ab \leq (a^2 + b^2)/2$ with

$$(A.5) \quad a^2 = (K_\beta/\lambda_{\min})n^{-\beta} \quad \text{and} \quad b^2 = E\|\theta_n\|^2/a^2,$$

we arrive at

$$\begin{aligned} E^{1/2}\|\theta_n\|^2 &\leq (K_\beta/2\lambda_{\min})n^{-\beta} + (\lambda_{\min}/2K_\beta)n^\beta E\|\theta_n\|^2 \\ &= (K_\beta/2\lambda_{\min})n^{-\beta} + (\lambda_{\min}/K_\beta)n^\beta EU(\theta_n). \end{aligned}$$

From the inequalities above and (A1), we have

$$(A.6) \quad E\|\theta'_n(\alpha_\theta(\theta_n) - E_n\psi_n)\| \leq (K_\beta^2/2\lambda_{\min})n^{-2\beta} + \lambda_{\min}EU(\theta_n).$$

By virtue of (A.3), (A.4), and (A.6), we obtain

$$\begin{aligned} (A.7) \quad EU(\theta_{n+1}) - EU(\theta_n) &\leq -\lambda_{\min}n^{-\gamma}EU(\theta_n) + K_1n^{-\gamma}EU(\theta_n) + K_1n^{-\gamma} + (K_\beta^2/2\lambda_{\min})n^{-\gamma-2\beta} \\ &\quad + 3K_\alpha^2n^{-2\gamma}(1 + 2EU(\theta_n)) + (3/2)K_\beta^2n^{-2\gamma-2\beta} \\ &\quad + (3/2)n^{-2\gamma}E\|\psi_n - E_n\psi_n\|^2, \end{aligned}$$

where $E\|\psi_n - E_n\psi_n\|^2 \leq K_\delta n^{-\delta}$ in accordance with (A2). Choose λ_0 such that $0 < \lambda_0 < \lambda_{\min}$. We may assume that n_0 has been chosen large enough such that

$$0 < \lambda_0 \leq \lambda_{\min} - 6K_\alpha^2n_0^{-\gamma} \quad \text{and} \quad 3n_0^{-\gamma} \leq \varepsilon_0/\lambda_{\min}.$$

Notice that there is a $K_2 > 0$ such that for $n \geq n_0$, $K_1n^{-\gamma} + 3K_\alpha^2n^{-2\gamma} \leq K_2n^{-\gamma}$. Using the inequalities above and iterating on (A.7) yields that for $n \geq n_0$,

$$\begin{aligned} (A.8) \quad EU(\theta_{n+1}) &\leq (1 - \lambda_0n^{-\gamma})EU(\theta_n) + K_1n^{-\gamma}EU(\theta_n) \\ &\quad + \frac{K_\beta^2}{\lambda_{\min}}n^{-\gamma-2\beta} + \frac{3K_\delta}{2}n^{-2\gamma-\delta} + K_2n^{-\gamma} \\ &= A_{n,n_0-1}EU(\theta_{n_0}) + K_1 \sum_{i=n_0}^n i^{-\gamma} A_{ni}EU(\theta_i) + \frac{K_\beta^2}{\lambda_{\min}} \sum_{i=n_0}^n i^{-\gamma-2\beta} A_{ni} \\ &\quad + \frac{3K_\delta}{2} \sum_{i=n_0}^n i^{-2\gamma-\delta} A_{ni} + K_2 \sum_{i=n_0}^n i^{-\gamma} A_{ni}, \end{aligned}$$

where

$$A_{nj} \stackrel{\text{def}}{=} \begin{cases} \prod_{k=j+1}^n (1 - \lambda_0 k^{-\gamma}), & j < n, \\ 1, & j = n. \end{cases}$$

Observe that for each fixed $j \neq n$,

$$\begin{aligned} (A.9) \quad |A_{nj}| &\leq \exp \left(-\lambda_0 \sum_{k=j+1}^n k^{-\gamma} \right) \\ &\leq \exp \left(\lambda_0 j^{-\gamma} - \lambda_0 \int_j^n x^{-\gamma} dx \right) \\ &= \exp \left(\lambda_0 j^{-\gamma} - \frac{\lambda_0(n^{1-\gamma} - j^{1-\gamma})}{1-\gamma} \right). \end{aligned}$$

As a consequence, the first term on the R.H.S. of (A.8) is $O(\exp(-\lambda_0 n^{1-\gamma}/(1-\gamma)))$. Consider now the third term in the last part of (A.8). By virtue of a partial summation, one has

$$(A.10) \quad \sum_{i=n_0}^n i^{-\gamma-2\beta} A_{ni} = n^{-2\beta} \sum_{i=n_0}^n i^{-\gamma} A_{ni} + \sum_{i=n_0}^{n-1} \left(\frac{1}{i^{2\beta}} - \frac{1}{(i+1)^{2\beta}} \right) \sum_{j=n_0}^i j^{-\gamma} A_{nj}.$$

Since

$$A_{nj} - A_{n,j-1} = A_{nj}(1 - (1 - \lambda_0 j^{-\gamma})) = \lambda_0 j^{-\gamma} A_{nj},$$

we have

$$\sum_{j=n_0}^i j^{-\gamma} A_{nj} = \lambda_0^{-1} \sum_{j=n_0}^i (A_{nj} - A_{n,j-1}) = \lambda_0^{-1} (A_{ni} - A_{n,n_0-1}).$$

Owing to (A.9),

$$\sum_{i=n_0}^{n-1} \left(\frac{1}{i^{2\beta}} - \frac{1}{(i+1)^{2\beta}} \right) A_{n,n_0-1} = O(A_{n,n_0-1}) = O(\exp(-n^{1-\gamma})).$$

Noticing the choice of n_0 and that

$$\frac{1}{n^{2\beta}} - \frac{1}{(n+1)^{2\beta}} = n^{-2\beta} (2\beta n^{-1} + O(n^{-2})) = 2\beta n^{-2\beta-1} + O(n^{-2\beta-2}),$$

and using (A.9) again, we have that

$$\begin{aligned} \sum_{i=n_0}^{n-1} \left(\frac{1}{i^{2\beta}} - \frac{1}{(i+1)^{2\beta}} \right) A_{ni} &= \sum_{i=n_0}^{n-1} (2\beta i^{-2\beta-1} + O(i^{-2\beta-2})) A_{ni} \\ &\leq \sum_{i=n_0}^n \varepsilon_0 i^{-2\beta-\gamma} A_{ni} + O(n^{-2\beta-1}). \end{aligned}$$

Combining this with (A.10) and observing that $\exp(-n^{1-\gamma}) = O(n^{-2\beta-1})$, we obtain that

$$\sum_{i=n_0}^n i^{-\gamma-2\beta} A_{ni} \leq \lambda_0^{-1} (A_{nn} - A_{n,n_0-1}) n^{-2\beta} + \varepsilon_0 \lambda_0^{-1} \sum_{i=n_0}^n i^{-\gamma-2\beta} A_{ni} + O(n^{-2\beta-1}),$$

and hence

$$(1 - \varepsilon_0/\lambda_0) \sum_{i=n_0}^n i^{-\gamma-2\beta} A_{ni} = \lambda_0^{-1} n^{-2\beta} + O(n^{-2\beta-1}).$$

By virtue of (A2), one has $\gamma + \delta > 1 - \gamma > 0$. Repeating the argument above, we also obtain

$$(A.11) \quad (1 - \varepsilon_0/\lambda_0) \sum_{i=n_0}^n i^{-2\gamma-\delta} A_{ni} = \lambda_0^{-1} n^{-\gamma-\delta} + O(n^{-\gamma-\delta-1}).$$

Owing to (A.8)–(A.11), there is a $K_3 > 0$ such that

$$(A.12) \quad EU(\theta_{n+1}) \leq \frac{K_\beta^2}{\lambda_{\min}\lambda_0(1-\varepsilon_0/\lambda_0)}n^{-2\beta} + \frac{3K_\delta}{2\lambda_0(1-\varepsilon_0/\lambda_0)}n^{-\gamma-\delta} \\ + O(n^{-2\beta-1} + n^{-\gamma-\delta-1}) + K_1 \sum_{i=n_0}^n i^{-\gamma} A_{ni} EU(\theta_i) + K_3.$$

By virtue of the Gronwall's inequality, there is a $K_4 > 0$ such that

$$EU(\theta_{n+1}) \leq K_4 \exp \left(K_1 \sum_{i=n_0}^n i^{-\gamma} A_{ni} \right),$$

and hence $\sup_n EU(\theta_n) \leq K_5$ for some $K_5 > 0$. This together with the w.p.1 convergence of $\theta_n \rightarrow 0$ implies that $EU(\theta_n) \rightarrow 0$.

Step 2. In view of (A5) and Step 1, $E[\theta'_n \alpha_\theta(\theta_n)] = E[\theta'_n H \theta_n] + o(EU(\theta_n))$ and $E\|\alpha_\theta(\theta_n)\|^2 \leq E[\theta'_n H' H \theta_n] + o(E[U(\theta_n)])$. As in the analysis in Step 1, however, we may assume that n_0 is chosen large enough so that $\|o(EU(\theta_n))\| \leq (1/2)\lambda_{\min} EU(\theta_n)$. In (A.5), choose $a^2 = (K_\beta/2\lambda_{\min})n^{-\beta}$. Substitute this into (A.3), and proceed exactly the same as before. However, this time, neither the term $K_1 n^{-\gamma} EU(\theta_n)$ nor $K_1 n^{-\gamma}$ appears. In lieu of (A.12), we obtain

$$EU(\theta_{n+1}) \leq \frac{K_\beta^2}{\lambda_{\min}\lambda_0(1-\varepsilon_0/\lambda_0)}n^{-2\beta} + \frac{3K_\delta}{2\lambda_0(1-\varepsilon_0/\lambda_0)}n^{-\gamma-\delta} + O(n^{-2\beta-1} + n^{-\gamma-\delta-1}).$$

Now, observe that it is possible to choose ε_0 and λ_0 such that $1/(\lambda_0 - \varepsilon_0) \leq (1 + \varepsilon)/\lambda_{\min}$; e.g., take $\lambda_0 = \lambda_{\min}/(1 + \varepsilon/2)$ and then $0 < \varepsilon_0 < \lambda_0 - \lambda_{\min}/(1 + \varepsilon)$. Since $\|\theta_n\|^2 = 2U(\theta_n)$, and since ε can be taken arbitrarily small in (A5), the first part of the theorem (for $1/2 < \gamma < 1$) follows.

The proof for the case $\gamma = 1$ is similar to that of $1/2 < \gamma < 1$. We only point out the difference below. Notice that (A.8) and (A.10) still hold with $\gamma = 1$. Equation (A.9) can be replaced by

$$\begin{aligned} |A_{nj}| &\leq \exp \left(\lambda_0/j - \lambda_0 \int_j^n x^{-1} dx \right) \\ &= \exp(\lambda_0/j - \lambda_0 \ln(n/j)) \\ &= \exp(\lambda_0/j) (j/n)^{\lambda_0}. \end{aligned}$$

We thus have

$$\begin{aligned} \sum_{i=n_0}^{n-1} i^{-2\beta-1} |A_{ni}| &\leq \exp(\lambda_0/n_0) \sum_{i=n_0}^{n-1} i^{-2\beta-1} (i/n)^{\lambda_0} \\ &\leq \exp(\lambda_0/n_0) n^{-\lambda_0} \int_{n_0}^n x^{\lambda_0-2\beta-1} dx \\ &\leq \exp(\lambda_0/n_0) (\lambda_0 - 2\beta)^{-1} n^{-2\beta}. \end{aligned}$$

Similarly,

$$\begin{aligned} \sum_{i=n_0}^{n-1} i^{-2-\delta} |A_{ni}| &\leq \exp(\lambda_0/n_0) \sum_{i=n_0}^{n-1} i^{-2-\delta} (i/n)^{\lambda_0} \\ &\leq \exp(\lambda_0/n_0) (\lambda_0 - \delta - 1)^{-1} n^{-\delta-1}. \end{aligned}$$

Since n_0 can be chosen sufficiently large, the proof of the second part follows. $\square$

Proof of Theorem 4.1. The proof is divided into three steps. In the first step, we establish an asymptotic equivalence, while in the second and third steps, we obtain the weak convergence by virtue of a result of Ethier and Kurtz [6], and compute the asymptotic covariance matrix. We first prove a lemma which gives an asymptotic equivalence.

LEMMA A.1. *Under the conditions of Theorem 4.1,*

$$W_n(t) = M_n(t) + o(1) \text{ for all } t \in [0, 1],$$

where

$$M_n(t) = \frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} \sum_{i=1}^{\lfloor nt \rfloor} i^{-1} D_{\lfloor nt \rfloor i} (E_i \psi_i - \psi_i),$$

D_{ni} is defined by

$$(A.13) \quad D_{ni} = \begin{cases} \prod_{j=i+1}^n (I - H/j), & n > i, \\ I, & n = i, \end{cases}$$

and $o(1) \xrightarrow{P} 0$ as $n \rightarrow \infty$, uniformly in t .

Proof of Lemma A.1. Direct computation leads to

$$\begin{aligned} \theta_{\lfloor nt \rfloor + 1} &= (I - H \lfloor nt \rfloor^{-1}) \theta_{\lfloor nt \rfloor} + \lfloor nt \rfloor^{-1} h(\theta_{\lfloor nt \rfloor}) \\ &\quad + \lfloor nt \rfloor^{-1} (\alpha_\theta(\theta_{\lfloor nt \rfloor}) - E_{\lfloor nt \rfloor} \psi_{\lfloor nt \rfloor}) + \lfloor nt \rfloor^{-1} (E_{\lfloor nt \rfloor} \psi_{\lfloor nt \rfloor} - \psi_{\lfloor nt \rfloor}) \\ &= D_{\lfloor nt \rfloor, n_0 - 1} \theta_{n_0} + \sum_{i=n_0}^{\lfloor nt \rfloor} i^{-1} D_{\lfloor nt \rfloor, i} h(\theta_i) \\ &\quad + \sum_{i=n_0}^{\lfloor nt \rfloor} i^{-1} D_{\lfloor nt \rfloor, i} (\alpha_\theta(\theta_i) - E_i \psi_i) + \sum_{i=n_0}^{\lfloor nt \rfloor} i^{-1} D_{\lfloor nt \rfloor, i} (E_i \psi_i - \psi_i), \end{aligned}$$

where n_0 is the same as in the proof of Theorem 3.1, and $h(\theta) = H\theta - \alpha_\theta(\theta)$. It then yields

$$\begin{aligned} \frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} \theta_{\lfloor nt \rfloor + 1} &= \frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} D_{\lfloor nt \rfloor, n_0 - 1} \theta_{n_0} + \frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} \sum_{i=n_0}^{\lfloor nt \rfloor} i^{-1} D_{\lfloor nt \rfloor, i} h(\theta_i) \\ &\quad + \frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} \sum_{i=n_0}^{\lfloor nt \rfloor} i^{-1} D_{\lfloor nt \rfloor, i} (\alpha_\theta(\theta_i) - E_i \psi_i) \\ (A.14) \quad &\quad + \frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} \sum_{i=n_0}^{\lfloor nt \rfloor} i^{-1} D_{\lfloor nt \rfloor, i} (E_i \psi_i - \psi_i). \end{aligned}$$

By virtue of the structure of $D_{k,j}$, it is easily seen that

$$\frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} D_{\lfloor nt \rfloor, n_0 - 1} \theta_{n_0} \xrightarrow{\text{w.p.1}} 0 \text{ as } n \rightarrow \infty.$$

Next, we obtain

$$\begin{aligned}
E \left\| \frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} \sum_{i=n_0}^{\lfloor nt \rfloor} i^{-1} D_{\lfloor nt \rfloor, i} h(\theta_i) \right\| &\leq K \frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} \sum_{i=n_0}^{\lfloor nt \rfloor} i^{-1} \|D_{\lfloor nt \rfloor, i}\| E \|\theta_i\|^2 \\
&\leq K \frac{1}{\sqrt{n}} \frac{1}{\lfloor nt \rfloor^{1+\delta/2}} \sum_{i=n_0}^{\lfloor nt \rfloor} \left(\frac{i}{\lfloor nt \rfloor} \right)^{\lambda_{\min} - 2 - \delta} \\
&\leq K n^{-\frac{1+\delta}{2}} \int_0^1 u^{\lambda_{\min} - 2 - \delta} du \xrightarrow{n \rightarrow \infty} 0
\end{aligned}$$

uniformly in t . Similarly,

$$E \left\| \frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} \sum_{i=n_0}^{\lfloor nt \rfloor} i^{-1} D_{\lfloor nt \rfloor, i} (\alpha_\theta(\theta_i) - E_i \psi_i) \right\| \xrightarrow{n \rightarrow \infty} 0 \quad \text{uniformly in } t.$$

In addition,

$$\frac{\lfloor nt \rfloor^{1+\delta/2}}{\sqrt{n}} \sum_{i=1}^{n_0-1} i^{-1} D_{\lfloor nt \rfloor, i} (E_i \psi_i - \psi_i) \xrightarrow{P} 0 \quad \text{as } n \rightarrow \infty, \text{ uniformly in } t.$$

Thus the lemma follows. $\square$

To complete the proof of Theorem 4.1, we apply Theorem 7.1.4 of Ethier and Kurtz [6]. First, observe that $M_n(\cdot)$ is a square integrable martingale (this follows from Theorem 3.1). Similar to Yin [41], it can be shown that

$$(A.15) \quad E \left(\sup_{t \in [0,1]} \|M_n(t) - M_n(t^-)\| \right) \xrightarrow{n \rightarrow \infty} 0$$

and the quadratic variation of $M_n(\cdot)$ is given by

$$\tilde{A}_n^{ij}(t) = \frac{\lfloor nt \rfloor^{2+\delta}}{n} \sum_{k=1}^{\lfloor nt \rfloor} k^{-2} (D_{\lfloor nt \rfloor, k} (E_k \psi_k - \psi_k))^i (D_{\lfloor nt \rfloor, k} (E_k \psi_k - \psi_k))^j,$$

where the superscripts i and j denote the i th and j th components, respectively, and $\tilde{A}_n^{ij}(\cdot)$ denotes the ij th entry of $\tilde{A}_n(\cdot)$. Furthermore, by virtue of Proposition 2.6.2 of Ethier and Kurtz [6], for $i, j = 1, 2, \dots, r$, $M_n^i(t)M_n^j(t) - \tilde{A}_n^{ij}(t)$ are martingales, where $M_n^i(\cdot)$ denotes the i th component of $M_n(\cdot)$.

Notice that $\tilde{A}_n^{ij}(\cdot)$ is nonnegative definite. By virtue of Condition (A7) and the square integrability of $M_n(\cdot)$,

$$\tilde{A}_n(t) = A_n(t) + o(1),$$

where

$$A_n(t) = \frac{\lfloor nt \rfloor^{2+\delta}}{n} \sum_{i=1}^{\lfloor nt \rfloor} i^{-2} D_{\lfloor nt \rfloor, i} R D'_{\lfloor nt \rfloor, i}$$

and $o(1) \xrightarrow{P} 0$ as $n \rightarrow \infty$, uniformly in t . In addition, $A_n(\cdot)$ is also nonnegative definite.

It is now clear that to derive the limit covariance matrix, we need only to look at $A_n^{ij}(\cdot)$. For a fixed $t \in [0, 1]$,

$$\begin{aligned} A_n(t) &= \frac{\lfloor nt \rfloor^{2+\delta}}{n} \sum_{i=1}^{\lfloor nt \rfloor} i^{-2-\delta} \exp(H \ln(i/\lfloor nt \rfloor)) R \exp(H' \ln(i/\lfloor nt \rfloor)) \\ &\quad + \frac{\lfloor nt \rfloor^{2+\delta}}{n} \sum_{i=1}^{\lfloor nt \rfloor} i^{-2-\delta} \left(D_{\lfloor nt \rfloor, i} R D'_{\lfloor nt \rfloor, i} \right. \\ &\quad \left. - \exp(H \ln(i/\lfloor nt \rfloor)) R \exp(H' \ln(i/\lfloor nt \rfloor)) \right). \end{aligned}$$

The last term on the R.H.S. above tends to 0 in probability as $n \rightarrow \infty$, uniformly in t . As for the first term, we have

$$\begin{aligned} &\frac{\lfloor nt \rfloor^{2+\delta}}{n} \sum_{i=1}^{\lfloor nt \rfloor} i^{-2-\delta} \exp(H \ln(i/\lfloor nt \rfloor)) R \exp(H' \ln(i/\lfloor nt \rfloor)) \\ &\xrightarrow{n \rightarrow \infty} t \int_0^1 u^{-2-\delta} \exp(H \ln u) R \exp(H' \ln u) du \\ &= t \int_0^\infty \exp((1+\delta)v) \exp(-Hv) R \exp(-H'v) dv \quad (\text{with } v = -\ln u) \\ &= t \int_0^\infty \exp(-\tilde{H}v) R \exp(-\tilde{H}'v) dv. \end{aligned}$$

Up to now, all the conditions in Theorem 7.1.4 of Ethier and Kurtz [6] are satisfied. The desired result then follows from that theorem. $\square$

Proof of Theorem 4.3. Similar to (A.14),

$$\begin{aligned} n^\beta \theta_{n+1} &= n^\beta D_{n, n_0-1} \theta_{n_0} + n^\beta \sum_{i=n_0}^n i^{-1} D_{ni} h(\theta_i) \\ &\quad + n^\beta \sum_{i=n_0}^n i^{-1} D_{ni} (\alpha_\theta(\theta_i) - E_i \psi_i) \\ &\quad + n^\beta \sum_{i=n_0}^n i^{-1} D_{ni} (E_i \psi_i - \psi_i). \end{aligned}$$

As in the derivation of Lemma A.1, upon using the condition $2\beta < 1 + \delta$ and the estimate $E\|\theta_n\|^2 = O(n^{2\beta})$, we have

$$(A.16) \quad n^\beta \theta_{n+1} = n^\beta \sum_{i=1}^n i^{-1} D_{ni} (\alpha_\theta(\theta_i) - E_i \psi_i) + o(1),$$

where $o(1) \xrightarrow{P} 0$ as $n \rightarrow \infty$.

Concentrating on the first term on the R.H.S. above, we obtain

$$\begin{aligned} &n^\beta \sum_{i=1}^n i^{-1} D_{ni} (\alpha_\theta(\theta_i) - E_i \psi_i) \\ &= -n^\beta \sum_{i=1}^n i^{-1} D_{ni} B_i \end{aligned}$$

$$\begin{aligned}
&= n^{-1} \sum_{i=1}^n (i/n)^{-1-\beta} D_{ni} \bar{B} - n^{-1} \sum_{i=1}^n (i/n)^{-1-\beta} D_{ni} [i^\beta B_i + \bar{B}] \\
&\xrightarrow{P} \int_0^1 u^{-1-\beta} \exp(H \ln u) \bar{B} du \quad \text{as } n \rightarrow \infty \\
&= \int_0^\infty \exp(-H_b v) \bar{B} dv \\
&= H_b^{-1} \bar{B}.
\end{aligned}$$

This yields the desired result. $\square$

Proof of Theorem 4.4. Owing to the fact that $\delta = 2\beta - 1$, we have

$$\begin{aligned}
n^{(1+\delta)/2} \theta_{n+1} &= n^{(1+\delta)/2} \sum_{i=1}^n i^{-1} D_{ni} (E_i \psi_i - \psi_i) \\
&\quad + n^{(1+\delta)/2} \sum_{i=1}^n i^{-1} D_{ni} (\alpha_\theta(\theta_i) - E_i \psi_i) + o(1),
\end{aligned}$$

where $o(1) \xrightarrow{P} 0$ as $n \rightarrow \infty$. Subtracting $H_b^{-1} \bar{B}$ from both sides of the equation above, noticing that $\beta = (1 + \delta)/2$ and applying Theorem 4.3, we establish that

$$n^{(1+\delta)/2} \theta_{n+1} - H_b^{-1} \bar{B} = n^{(1+\delta)/2} \sum_{i=1}^n i^{-1} D_{ni} (E_i \psi_i - \psi_i) + o(1),$$

where $o(1) \xrightarrow{P} 0$ as $n \rightarrow \infty$. This, with Theorem 4.1, completes the proof. $\square$

Appendix B. Details on convergence for specific setups. This appendix provides some technical details on the convergence of SA for the specific setups studied in section 5.

Finite-horizon models (Example 5.1). Let the assumptions made in Example 5.1 hold and assume that $\text{Var}[X] \rightarrow \sigma^2$ as $\theta \rightarrow 0$. In the case of FD with independent random numbers, we have the following (see L'Ecuyer and Perron [27] or Zazanis and Suri [46]): For FDF, if α is twice continuously differentiable at $\theta^* = 0$ with second derivative $H = \lambda_{\min} > 0$, then $n^\nu B_n \rightarrow H\kappa_c/2$ and $n^{p-2\nu} V_n \rightarrow 2\kappa_N^{-1}\kappa_c^{-2}\sigma^2$. For FDC, if α is three times continuously differentiable at 0 and H_3 denotes the third derivative, then $n^{2\nu} B_n \rightarrow H_3\kappa_c^2/6$ and $n^{p-2\nu} V_n \rightarrow \kappa_N^{-1}\kappa_c^{-2}\sigma^2/2$. Assumptions (A7) and (A9) then hold for $\delta = p - 2\nu$, $\beta = \nu$ for FDF, and $\beta = 2\nu$ for FDC. We assume that (3.2) holds and that θ is restricted (e.g., by projection if necessary) to a convex region of G where the second and the third derivatives of α and the variance of X are bounded uniformly in θ . Then, (A1) and (A2) also hold, and we have all the ingredients to apply Theorem 3.1 as well as the results of section 4.

According to Corollary 4.5, and assuming that H is large enough to satisfy (A6) and (A8), the optimal values of ν and p must satisfy $1 + \delta = 2\beta$; i.e., $\nu = (1 + p)/4$ for FDF and $\nu = (1 + p)/6$ for FDC. This yields

$$C_n^{\eta/2} \theta_n \xrightarrow{d} \kappa_b^{\eta/2} N(-\mu, \Sigma),$$

where $\eta = 1/2$, $\mu = H\kappa_c/(2(H - \nu))$, and $\Sigma = \sigma^2/(\kappa_N\kappa_c^2(H - \nu))$ for FDF, while $\eta = 2/3$, $\mu = H_3\kappa_c^2/(6(H - 2\nu))$, and $\Sigma = \sigma^2/(4\kappa_N\kappa_c^2(H - 2\nu))$ for FDC. The

asymptotic rate η does not depend on p , but the asymptotic MSE constant

$$(B.1) \quad K_{\text{MSE}} = \kappa_b^\eta (\mu^2 + \Sigma) \stackrel{\text{w.p.1}}{=} \lim_{n \rightarrow \infty} C_n^\eta E[\|\theta_n\|^2]$$

does. Note that $p = 0$ is not necessarily the optimal choice. If we restrict ourselves to $p = 0$, the constant K_{MSE} is minimized for FDF by choosing κ_N and κ_c such that $\kappa_N \kappa_c^4 = 4(H - 1/4)\sigma^2 H^{-2}$, which yields $K_{\text{MSE}} = \sigma \kappa_T^{1/2} (H - 1/4)^{-3/2}$. For FDC, again for $p = 0$, the constant K_{MSE} is minimized by choosing $\kappa_N \kappa_c^6 = (9/2)(H - 1/3)\sigma^2 H_3^{-2}$, which yields $K_{\text{MSE}} = (\sigma^4 H_3^2 (H - 1/3)^{-4} \kappa_T^2 / 9)^{1/3} (2^{1/3} + 2^{-2/3})/4$.

In the case where common random numbers are used and the CRN2 conditions hold, let $\sigma_P^2 = \text{Var}[X_\theta(0)]$. Then, for both (5.1) and (5.2), $\text{Var}[\psi]$ converges to σ_P^2 as $c \rightarrow 0$ and $\theta \rightarrow 0$. This gives $n^p V_n \rightarrow \kappa_N^{-1} \sigma_P^2$ and $\Sigma = \sigma_P^2 / (2\kappa_N(H - \nu))$. The bias satisfies $n^\nu B_n \rightarrow H\kappa_c/2$ for FDF and $n^{2\nu} B_n \rightarrow H_3 \kappa_c^2/6$ for FDC. Under the conditions given in Table 5.1, we obtain $\eta = 1$. More precisely, under those conditions

$$C_n^{1/2} \theta_n \xrightarrow{d} \kappa_b^{1/2} N(-\mu, \Sigma),$$

where $\kappa_b = \kappa_T \kappa_N / (p + 1)$, $\Sigma = \sigma_P^2 / (2\kappa_N(H - (1 + p)/2))$, $\mu = I[2\nu = p + 1] \cdot H\kappa_c / (2(H - \nu))$ for FDF, $\mu = I[4\nu = p + 1] \cdot H_3 \kappa_c^2 / (6(H - 2\nu))$ for FDC, and $I[A]$ denotes the indicator function which takes the value 1 when the underlying variable belongs to A and is 0 otherwise. It is best to take ν such that $2\beta > p + 1$ because it cancels out the bias term μ . Then, $K_{\text{MSE}} = \kappa_b \Sigma$ can be minimized by taking $p = H - 1$, which yields $K_{\text{MSE}} = \sigma_P^2 \kappa_T H^{-2}$. Recall that the value of H can also be changed by rescaling the problem.

In the CRN1 case, we have $\mu = H\kappa_c / (2(H - \nu))$ and $\Sigma = \sigma_D^2 / (2\kappa_N(H - \nu))$ for FDF, while $\mu = H_3 \kappa_c^2 / (6(H - 2\nu))$ and $\Sigma = \sigma_D^2 / (4\kappa_N(H - \nu))$ for FDC. Again, the value of p that minimizes the asymptotic MSE constant is not necessarily 0.

Steady-state average cost (Example 5.3). The expressions for V_n and B_n here are related to those of Example 5.1, modified as follows: Multiply V_n by $\Theta(t_n^{-1}) = \Theta(n^{-q})$ and add $\Theta(t_n^{-1})$ to $\|B_n\|$. It is reasonable to suppose that (5.3) holds with N_n replaced by $N_n t_n$, which implies that $n^{-p-q-1} C_n \rightarrow \kappa_b$ w.p.1 for some positive constant κ_b .

Under those assumptions (and those made in Example 5.1 for the finite horizon), we obtain the following. For the FD estimators with independent random numbers, we have $V_n = \Theta(t_n^{-1} N_n^{-1} c_n^{-2}) = \Theta(n^{2\nu-p-q})$, for both FDF and FDC, $\|B_n\| = \Theta(t_n^{-1} + c_n) = \Theta(n^{-(q \wedge \nu)})$ for FDF, and $\|B_n\| = \Theta(t_n^{-1} + c_n^2) = \Theta(n^{-(q \wedge 2\nu)})$ for FDC. With common random numbers, B_n is the same, while $V_n = \Theta(t_n^{-1} N_n^{-1} c_n^{-1}) = \Theta(n^{\nu-p-q})$ for CRN1 and $V_n = \Theta(t_n^{-1} N_n^{-1}) = \Theta(n^{-p-q})$ for CRN2. For IPA, both $\|B_n\|$ and V_n are $\Theta(t_n^{-1} N_n) = \Theta(n^{-p-q})$.

For the LR estimator, we assume here that $V_n = \Theta(t_n) = \Theta(n^q)$ and $\|B_n\| = \Theta(t_n^{-1}) = \Theta(n^{-q})$. For CLR, we have $V_n = \Theta(1)$ and $\|B_n\| = \Theta(t_n^{-1}) = \Theta(n^{-q})$. From that, the optimal values given in Table 5.2 are easily obtained.

Infinite-horizon discounted model (Example 5.6). Let $t_n / \ln n \rightarrow \kappa_t$ as $n \rightarrow \infty$, where $\kappa_t \in [0, \infty]$. We want to show that κ_t must be finite and large enough (as specified in Table 5.3) to maximize $\tilde{\eta}$ and that the convergence rate actually depends on the product $\rho \kappa_t$. Recall that the derivative estimator ψ_n is the average of $N_n = \kappa_N n^p$ i.i.d. replications of either $X_\theta(\theta_n, t_n)$ (for IPA) or of a finite difference (for FD).

For IPA, we have $n^p V_n \rightarrow \sigma^2 \equiv R$ as $\theta_n \rightarrow 0$, $B_n \sim \kappa_\beta \exp(-\rho t_n) \sim \kappa_\beta n^{-\rho \kappa_t}$, and the results of sections 3 and 4 apply with $\beta = \rho \kappa_t$ and $\delta = p$ (assuming that $\max(1 + 2\rho \kappa_t, 2 + p) < \lambda_{\min}$). This yields $E[\|\theta_n\|^2] = \Theta(n^{-\kappa})$ with $\kappa = 2\rho \kappa_t \wedge (p + 1)$. We also have (w.p.1) $T_n \sim \kappa_T \kappa_N \kappa_t n^p \ln n$, $E[C_n] \sim \kappa_T \kappa_N \kappa_t n^{(p+1)} \ln n / (p + 1)$, and

$(\ln E[C_n])/E[C_n] \sim (p+1)^2(\kappa_T\kappa_N\kappa_t)^{-1}n^{-(p+1)}$. For $\kappa_t \geq (p+1)/(2\rho)$, this gives $E[\|\theta_n\|^2] = \Theta(n^{-p-1}) = \Theta((\ln E[C_n])/E[C_n])$. Upon a closer examination, we also see that the latter MSE increases linearly with κ_t and that the bias term disappears ($2\beta > 1 + \delta$) when $\kappa_t > (p+1)/(2\rho)$. If $\kappa_t = \infty$ (i.e., t_n increases faster than $\Theta(\ln n)$), then the MSE must increase faster than $\Theta((\ln E[C_n])/E[C_n])$. For $\kappa_t < (p+1)/(2\rho)$ (including $\kappa_t = 0$), we obtain $E[\|\theta_n\|^2] = \Theta(n^{-2\rho\kappa_t})$, which converges at a slower rate than $\Theta(n^{-p-1})$. Therefore, the optimal strategy is to take κ_t equal to (or slightly larger than) $(p+1)/(2\rho)$. We just showed that the optimal rate in terms of $E[C_n]$ is obtained if and only if $t_n/\ln n \rightarrow \kappa_t$ where κ_t is a large enough but finite constant.

Similar analysis can be carried out when $\alpha_\theta(\theta, t)$ is estimated via finite differences using $X(\theta, t)$ instead of $X(\theta)$ in each of the setups of Example 5.1. For FDc with independent random numbers, for example, one has $\|B_n\| = \Theta(n^{-2\nu} + e^{-\rho t_n}) = \Theta(n^{-(2\nu \wedge \rho\kappa_t)})$ and $V_n = \Theta(n^{2\nu-p})$. Take $\nu = (p+1)/6$ and $\kappa_t \geq (p+1)/(3\rho)$. Then, $\|B_n\| = \Theta(n^{-(p+1)/3})$, $V_n = \Theta(n^{(p+1)/3})$, and $E\|\theta_n\|^2 = \Theta(n^{-2(p+1)/3}) = \Theta(((E[C_n])^{-1} \ln E[C_n])^{2/3})$. It is not hard to see that one cannot improve upon that convergence rate and that the rate is lower if $\kappa_t < (p+1)/(3\rho)$ or if $t_n/\ln n \rightarrow \infty$. The results of the other lines of Table 5.3 can be obtained in a similar fashion; we leave out the details.

Acknowledgments. We thank Associate Editor John Birge, the referees, and Qian-Yu Tang for their constructive comments.

REFERENCES

- [1] S. ANDRADÓTTIR, *A stochastic approximation algorithm with varying bounds*, Oper. Res., 43 (1995), pp. 1037–1048.
- [2] F. AZADIVAR AND J. TALAVAGE, *Optimization of stochastic simulation models*, Math. and Comput. in Simulation, XXII (1980), pp. 231–241.
- [3] J. A. BATHER, *Stochastic approximation: A generalization of the Robbins-Monro procedure*, in Proc. 4th Prague Symposium Asymptotic Statist., P. Mandl and M. Hušková, eds., 1989, pp. 13–27.
- [4] H. F. CHEN AND Y. M. ZHU, *Stochastic approximation procedure with randomly varying truncations*, Scientia Sinica, 29 (1986), pp. 914–926.
- [5] K. L. CHUNG, *On a stochastic approximation method*, Annals of Mathematical Statistics, 25 (1954), pp. 463–483.
- [6] S. N. ETHIER AND T. G. KURTZ, *Markov Processes: Characterization and Convergence*, John Wiley, New York, 1986.
- [7] V. FABIAN, *On asymptotic normality in stochastic approximation*, Annals of Mathematical Statistics, 39 (1968), pp. 1327–1332.
- [8] B. L. FOX AND P. W. GLYNN, *Simulating discounted costs*, Management Science, 35 (1989), pp. 1297–1315.
- [9] P. GLASSERMAN, *Gradient Estimation Via Perturbation Analysis*, Kluwer Academic Publishers, Norwell, MA, 1991.
- [10] P. GLASSERMAN AND D. D. YAO, *Some guidelines and guarantees for common random numbers*, Management Science, 38 (1992), pp. 884–908.
- [11] P. W. GLYNN, *A GSMP formalism for discrete event systems*, Proc. of the IEEE, 77 (1989), pp. 14–23.
- [12] P. W. GLYNN, *Likelihood ratio gradient estimation for stochastic systems*, Communications of the ACM, 33 (1990), pp. 75–84.
- [13] P. W. GLYNN AND P. HEIDELBERGER, *Bias properties of budget constrained simulations*, Oper. Res., 38 (1990), pp. 801–814.
- [14] P. W. GLYNN AND P. L'ECUYER, *On the existence and estimation of performance measure derivatives for stochastic recursions*, in Proc. of the 11th Internat. Conf. on Anal. and Optim. of Systems, Lecture Notes in Control and Inform. Sci., 199, Springer-Verlag, New York, 1994, pp. 429–435.
- [15] P. W. GLYNN, P. L'ECUYER, AND M. ADÈS, *Gradient estimation for ratios*, in Proc. of the 1991 Winter Simulation Conference, IEEE Press, Piscataway, NJ, December 1991, pp. 986–993.

- [16] P. W. GLYNN AND W. WHITT, *The asymptotic efficiency of simulation estimators*, Oper. Res., 40 (1992), pp. 505–520.
- [17] A. HAURIE, P. L'ECUYER, AND CH. VAN DELFT, *Monte-Carlo optimization of parametrized policies in a class of piecewise deterministic control systems arising in manufacturing flow control*, Discrete Event Dynamic Systems: Theory and Appl., 4 (1994), pp. 87–111.
- [18] J. KOMLÓS AND P. RÉVÉZ, *A modification of the Robbins–Monro process*, Studia Sci. Math. Hungar., 8 (1973), pp. 329–340.
- [19] H. J. KUSHNER, *Approximation and Weak Convergence Methods for Random Processes*, MIT Press, Cambridge, MA, 1984.
- [20] H. J. KUSHNER AND D. S. CLARK, *Stochastic Approximation Methods for Constrained and Unconstrained Systems*, Appl. Math. Sci., 26, Springer-Verlag, Berlin, New York, 1978.
- [21] H. J. KUSHNER AND H. HUANG, *Rates of convergence for stochastic approximation type algorithms*, SIAM J. Control Optim., 17 (1979), pp. 607–617.
- [22] H. J. KUSHNER AND F. J. VÁZQUEZ-ABAD, *Stochastic approximation methods for systems over an infinite horizon*, SIAM J. Control Optim., 34 (1996), pp. 712–756.
- [23] H. J. KUSHNER AND J. YANG, *Stochastic approximation with averaging of the iterates: Optimal asymptotic rate of convergence for general processes*, SIAM J. Control Optim., 31 (1993), pp. 1045–1062.
- [24] H. J. KUSHNER AND G. YIN, *Stochastic Approximation Algorithms and Applications*, Springer-Verlag, Berlin, New York, 1997.
- [25] P. L'ECUYER, *An overview of derivative estimation*, in Proc. of the 1991 Winter Simulation Conference, IEEE Press, Piscataway, NJ, December 1991, pp. 207–217.
- [26] P. L'ECUYER, *Convergence rates for steady-state derivative estimators*, Ann. Oper. Res., 39 (1992), pp. 121–136.
- [27] P. L'ECUYER AND G. PERRON, *On the convergence rates of IPA and FDC derivative estimators*, Oper. Res., 42 (1994), pp. 643–656.
- [28] P. L'ECUYER, N. GIROUX, AND P. W. GLYNN, *Stochastic optimization by simulation: Numerical experiments with the M/M/1 queue in steady-state*, Management Science, 40 (1994), pp. 1245–1261.
- [29] P. L'ECUYER AND P. W. GLYNN, *Stochastic optimization by simulation: Convergence proofs for the GI/G/1 queue in steady-state*, Management Science, 40 (1994), pp. 1562–1578.
- [30] P. L'ECUYER AND P. W. GLYNN, *Uniform convergence of truncated-horizon estimators for parametrized regenerative systems*, manuscript, 1996.
- [31] S. P. MEYN AND R. L. TWEEDIE, *Markov Chains and Stochastic Stability*, Springer-Verlag, Berlin, New York, 1993.
- [32] M. B. NEVEL'SON AND R. Z. KHASMINSKII, *Stochastic Approximation and Recursive Estimation*, Transl. Math. Monogr., 47, American Mathematical Society, Providence, RI, 1976.
- [33] B. T. POLYAK, *New method of stochastic approximation type*, Avtomat. i Telemekh., 7 (1990), pp. 937–946. Translation in Automat. Remote Control, 51 (1991), pp. 937–946.
- [34] B. T. POLYAK AND A. B. JUDITSKY, *Acceleration of stochastic approximation by averaging*, SIAM J. Control Optim., 30 (1992), pp. 838–855.
- [35] R. Y. RUBINSTEIN AND A. SHAPIRO, *Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization by the Score Function Method*, John Wiley, New York, 1993.
- [36] R. RUPPERT, *Almost-sure approximation to the Robbins–Monro and Kiefer–Wolfowitz processes with dependent noise*, Annals Probab., 10 (1982), pp. 178–187.
- [37] R. RUPPERT, *Handbook in sequential analysis*, in Stochastic Approximation, B. K. Ghosh and P. K. Sen, eds., Marcel Dekker, New York, 1991, pp. 503–529.
- [38] R. SCHWABE, *Stability results for smoothed stochastic approximation procedures*, Tech. rep. Preprint Nr. A-92-14, Fachbereich Mathematik, Freie Universität Berlin, Berlin, Germany, 1992.
- [39] K. SIGMAN AND R. W. WOLFF, *A review of regenerative processes*, SIAM Rev., 35 (1993), pp. 269–288.
- [40] M. T. WASAN, *Stochastic Approximation*, Cambridge University Press, London, 1969.
- [41] G. YIN, *A stopping rule for the Robbins–Monro method*, J. Optim. Theory Appl., 67 (1990), pp. 151–173.
- [42] G. YIN, *On extensions of Polyak's averaging approach to stochastic approximation*, Stochastics, 36 (1992), pp. 245–264.
- [43] G. YIN, H. YAN, AND X. LOU, *On a class of stochastic optimization algorithms with applications to manufacturing models*, in Proceedings of the Third Model-Oriented Data Analysis, H. P. W. G. Müller and A. A. Zhigljavsky, eds., Physica-Verlag, Heidelberg, 1993, pp. 213–226.

- [44] G. YIN AND K. YIN, *Asymptotically optimal rate of convergence of smoothed stochastic recursive algorithms*, Stochastics, 47 (1994), pp. 21–46.
- [45] G. YIN AND Y. M. ZHU, *Almost sure convergence of stochastic approximation algorithms with non-additive noise*, Internat. J. Control, 47 (1989), pp. 1361–1376.
- [46] M. A. ZAZANIS AND R. SURI, *Convergence rates of finite-difference sensitivity estimates stochastic systems*, Oper. Res., 41 (1993), pp. 694–703.

THE SEQUENTIAL KNAPSACK POLYTOPE*

Y. POCHET[†] AND R. WEISMANTEL[‡]

Abstract. In this paper we describe the convex hull of all solutions of the integer bounded knapsack problem in the special case when the weights of the items are divisible. The corresponding inequalities are defined via an inductive scheme that can also be used in a more general setting.

Key words. integer programming, knapsack polytope, knapsack problem, linear programming formulation, separation

AMS subject classification. 90C

PII. S1052623495285217

1. Introduction. In this paper we deal with the integer bounded knapsack problem defined on the set of items $N = \{1, \dots, n\}$.

$$\begin{aligned}
 \max \quad & \sum_{i=1}^n \gamma_i x_i, \\
 (SKP) \quad & \text{s.t.} \quad \sum_{i=1}^n a_i x_i \leq a_0, \\
 & x_i \in \{0, \dots, s_i\} \quad \text{for } i = 1, \dots, n,
 \end{aligned}$$

where $0 < a_1 \leq a_2 \leq \dots \leq a_n \leq a_0$, $a_0, a_i, s_i \in \mathbb{N}$ for $i = 1, \dots, n$, and the numbers a_i are divisible, i.e., $a_i/a_{i-1} \in \mathbb{N}$ for $i = 2, \dots, n$. In this case we say that the set of items, the knapsack inequality, and the associated knapsack problem have the *divisibility property*. It is also called the *sequential knapsack problem* (see [1]). Whenever we are given a knapsack problem having the divisibility property, we will assume without loss of generality that $a_1 = 1$.

Our main result is the construction of a system of linear inequalities that describes the convex hull of all solutions in this special case.

For 30 years the knapsack polytope has been of particular interest to researchers in polyhedral combinatorics. There are several reasons for this: One is the increasing number of applications in areas such as circuit design, telecommunications, vehicle routing, and scheduling that involve the knapsack problem as a subproblem. In order to apply polyhedral methods to such complex problems, a good understanding of the knapsack polytope is important. Another reason is that the knapsack problem is the “easiest” example of a “number dependent” problem. A slight change of the weights of the items might change the inequalities that describe the polytope drastically. Therefore, it is important to understand “general principles” behind the construction of valid inequalities. Examples in this direction are, for instance, Gomory cutting planes [2], covers [12], $(1, k)$ -configurations [8], the concept of lifting [7], the weight

* Received by the editors April 24, 1995; accepted for publication (in revised form) January 8, 1997. This research was partially supported by Science Program SC1-CT91-620 of the EEC and contract ERB CHBGCT 920167 of the EEC Human Capital and Mobility Program.

<http://www.siam.org/journals/siopt/8-1/28521.html>

[†] CORE and IAG, Université Catholique de Louvain, B-3001 Louvain, Belgium.

[‡] Konrad-Zuse-Zentrum für Informationstechnik Berlin, D-14195 Berlin, Germany (weismantel@zib-berlin.de).

reduction principle [10], or inequalities based on the Hilbert basis of a cone of exchange vectors [11].

We present an inductive scheme to construct valid inequalities for the knapsack polytope. This scheme can be applied to derive the complete description of the knapsack polytope when the set of items has the divisibility property.

The special case of the knapsack problem with the divisibility property has been studied in the literature by several authors. Hartmann and Olmstead [4] give an $O(n \log n)$ algorithm for solving *SKP*. Its bottleneck operation is the sorting of the ratios $\frac{\gamma_i}{a_i}$, $i \in N$. The case of the sequential knapsack problem when $s_i = \infty$ for all $i \in N$ has been considered by Marcotte [6]. She showed that an optimal solution can be found in linear time and that the associated knapsack polytope is described by a polynomial number of inequalities. Pochet and Wolsey [9] give an explicit description of the $\geq$ -knapsack polytope with the divisibility property when there are no bounds on the variables (i.e., $s_i = \infty$ for all $i \in N$). We also refer to these two papers for applications of the sequential knapsack problem in local area networking and cutting stock. Note that in this paper we deal with finite bounds s_i on the variables and therefore we do not need to consider the $\geq$ -knapsack polytope with the divisibility property because, by complementing the variables, it is easily shown to be equivalent to *SKP*.

In section 2 we present a transformation of a given sequential knapsack problem to a special one such that in terms of optimization both formulations are equivalent. It turns out that the optimal solutions of such a transformed sequential knapsack problem can be nicely decomposed. This is discussed in section 3. Our main result is presented in section 4. There we define an inductive scheme to generate valid inequalities for the sequential knapsack problem. For an arbitrary objective function we construct an inequality using the inductive scheme whose induced face contains the set of all optimal solutions. This suffices to show that our inductive class of inequalities describes the sequential knapsack polytope. How inequalities defined via our inductive scheme can be interpreted combinatorially is the issue of section 5. We briefly discuss in section 6 how the polyhedral description of the sequential knapsack problem can be used to obtain valid inequalities for the general 0/1 knapsack problem. We conclude in section 7 by briefly giving some results and some ideas for the separation problem associated with the sequential knapsack polytope.

Throughout the paper we use the following notation.

For $v \in \mathbb{R}$, we set $v^+ := \max\{v, 0\}$, $\lceil v \rceil := \min\{j \in \mathbb{N} : j \geq v\}$, and $\lfloor v \rfloor := \max\{j \in \mathbb{N} : j \leq v\}$.

The constraint $\sum_{i=1}^n a_i x_i \leq a_0$ is called the *knapsack inequality*. The number $a_i \in \mathbb{N}$ is termed the *weight* of item i , and $a_0 \in \mathbb{N}$ is called the *knapsack capacity*. A nonnegative integer vector x that satisfies the knapsack constraint and the upper bound constraints is called *feasible*.

For a polytope $\mathcal{P}$, the face F_c induced by the valid inequality $c^T x \leq \gamma$ is defined by $F_c = \{x \in \mathcal{P} \mid c^T x = \gamma\}$. If the dimension of F_c is one less than the dimension of $\mathcal{P}$, F_c is called a facet. An inequality that induces this facet is called facet defining. The inequalities $x_i \leq s_i$, $i \in N$ and $x_i \geq 0$, $i \in N$ are called *trivial*. For real numbers τ_j , $j = 1, \dots, n$, we define $\sum_{j=v}^w \tau_j := 0$ if $v > w$, and for $I \subseteq \{1, \dots, n\}$ we use the notation $\tau(I) := \sum_{i \in I} \tau_i$ with $\tau(\emptyset) = 0$.

2. A transformation. In this section we present a transformation of the given sequential knapsack problem to a special sequential knapsack problem that satisfies certain requirements. We show that in terms of optimization both formulations are

“equivalent.” We start by introducing the notion of *blocks*.

DEFINITION 2.1. Let $B := \{i_1, \dots, i_l\}$, $i_1 < \dots < i_l$, be a subset of items. B is called a block if, for every $j \in \{2, \dots, l\}$, $a_{i_j} \leq a_{i_1} + \sum_{v=1}^{j-1} s_{i_v} a_{i_v}$ holds. The number $u_B := \sum_{v=1}^l s_{i_v} a_{i_v} / a_{i_1}$ is called the multiplicity of block B .

The following property is easily verified for blocks.

PROPOSITION 2.2. Let $B = \{i_1, \dots, i_l\}$ be a block. For every number $\tau \in \{a_{i_1}, 2a_{i_1}, \dots, \sum_{v=1}^l s_{i_v} a_{i_v}\}$ there exists a subset $W \subseteq B$ and integers $0 < \lambda_k \leq s_k$ for all $k \in W$ such that $\sum_{k \in W} \lambda_k a_k = \tau$.

In the following we replace a block B by a single item B with weight a_{i_1} , multiplicity (upper bound) u_B , and objective function coefficient γ_{i_1} .

Let $B_1, \dots, B_m$ be a partition of N into blocks, and, for $w = 1, \dots, m$, denote by f_w , c_w , and u_w the weight, objective function coefficient, and multiplicity of block B_w , respectively. The knapsack problem associated with the set of blocks reads as follows.

$$\begin{aligned}
 (MSKP) \quad & \max \quad \sum_{w=1}^m c_w z_w, \\
 & \text{s.t.} \quad \sum_{w=1}^m f_w z_w \leq a_0, \\
 & \quad z_w \in \{0, \dots, u_w\} \quad \text{for } w = 1, \dots, m.
 \end{aligned}$$

From the construction of the blocks it is clear that $MSKP$ is a sequential knapsack problem ($MSKP$ stands for modified sequential knapsack problem). We will show that for a special partition of N into blocks (namely the partition into maximal blocks), the optimization problems SKP and $MSKP$ are equivalent.

Before that, we consider first the relation between the feasible solutions of these two problems and show that there is a many to one correspondence between the feasible solutions of the original problem (SKP) and the feasible solutions of its modified version ($MSKP$). For ease of notation we assume that $f_1 \leq f_2 \leq \dots \leq f_m$, and in case $f_w = f_{w+1}$, then $c_w \geq c_{w+1}$ holds. By $\mathcal{P}_{SKP}$ and $\mathcal{P}_{MSKP}$ we denote the convex hull of all feasible vectors of problems SKP and $MSKP$, respectively. From the fact that $a_n \leq a_0$, and $f_m \leq a_0$, it is obvious that both $\mathcal{P}_{SKP}$ and $\mathcal{P}_{MSKP}$ are full-dimensional polyhedra. Therefore, there exists a unique minimal system of linear inequalities that describes each of these polytopes.

Let $z \in \mathbb{Z}^m$ be a feasible solution of $MSKP$, i.e., $0 \leq z_w \leq u_w$, z_w integer for all $w = 1, \dots, m$, and $\sum_{w=1}^m f_w z_w \leq a_0$. By Definition 2.1 and Proposition 2.2, there exist integers $0 \leq \lambda_1 \leq s_1, \dots, 0 \leq \lambda_n \leq s_n$ such that, for every block B_w , the equation $\sum_{j \in B_w} a_j \lambda_j = f_w z_w$ holds. In fact, for all integers λ_j , $j = 1, \dots, n$ satisfying $0 \leq \lambda_j \leq s_j$, $j = 1, \dots, n$, and $\sum_{j \in B_w} a_j \lambda_j = f_w z_w$, $w = 1, \dots, m$, the vector $x \in \mathbb{Z}^n$ defined by $x_j = \lambda_j$ is feasible for problem SKP .

Conversely, with every vector $x \in \mathbb{Z}^n$ that is feasible for problem SKP , we associate a vector $z \in \mathbb{Z}^m$ by setting $z_w := \sum_{j \in B_w} (a_j / f_w) x_j$ for all $w = 1, \dots, m$. Then, $\sum_{i=1}^n a_i x_i = \sum_{w=1}^m f_w z_w$.

It follows that an integer vector z with $z_w \in \{0, \dots, u_w\}$ for $w = 1, \dots, m$ is feasible for $MSKP$ if and only if there exist feasible vectors of SKP with the same total weight as z .

Now suppose that $\sum_{w=1}^m b_w z_w \leq b_0$ is a valid inequality for the polytope $\mathcal{P}_{MSKP}$. By setting $\beta_i := b_w a_i / f_w$ if item i belongs to block B_w , the inequality $\sum_{i=1}^n \beta_i x_i \leq b_0$

is valid for $\mathcal{P}_{SKP}$. This statement follows from the fact that if x is feasible for SKP , then $z = (z_1, \dots, z_m)^T$ with coefficients $z_w = \sum_{j \in B_w} (a_j / f_w) x_j$ is feasible for $MSKP$ and satisfies $\sum_{i \in B_w} \beta_i x_i = b_w z_w$. This shows that valid inequalities for $\mathcal{P}_{MSKP}$ can be transformed into valid inequalities for $\mathcal{P}_{SKP}$.

Now we consider the equivalence between the optimization problems SKP and $MSKP$ and focus on a special partition of the set N into blocks $B_1, \dots, B_m$. For an item i of SKP , its *gain per unit* is defined as $\frac{\gamma_i}{a_i}$. Let $g_1, \dots, g_v$ denote the different values of gains per unit for all items of SKP (clearly, $v \leq n$) and define V_k as the set of items with gain per unit equal to g_k , $k = 1, \dots, v$. We partition each set $V_k = \{i \in N : \gamma_i / a_i = g_k\}$, $k = 1, \dots, v$, into blocks $B_1^k, \dots, B_{n_k}^k$ such that $\cup_{j \in I} B_j^k$ is not a block anymore for all $I \subseteq \{1, \dots, n_k\}$ with $|I| \geq 2$. Let $B_1, \dots, B_m$, with $m = \sum_{k=1}^v n_k$, denote the final blocks constructed this way. Each block B_w , $w = 1, \dots, m$ is called a *maximal block*, and, by definition, all items belonging to the same block B_w have the same gain per unit.

Example 2.3. Consider the instance of the sequential knapsack problem

$$\begin{aligned} \max \quad & x_1 + 3x_2 + 6x_3 + 18x_4 + 6x_5 + 50x_6 + 200x_7, \\ & x_1 + 5x_2 + 10x_3 + 30x_4 + 30x_5 + 120x_6 + 360x_7 \leq 396 \end{aligned}$$

with upper bounds $s_1 = 4$, $s_2 = 4$, $s_3 = 20$, $s_4 = 4$, $s_5 = 2$, $s_6 = 1$, and $s_7 = 1$. The set of items is partitioned into the five maximal blocks $V_1 = B_1 = \{1\}$, $V_2 = B_2 = \{2, 3, 4\}$, $V_3 = B_3 = \{5\}$, $V_4 = B_4 = \{6\}$, and $V_5 = B_5 = \{7\}$. After transformation we obtain the instance of the (modified) sequential knapsack problem

$$\begin{aligned} \max \quad & z_1 + 3z_2 + 6z_3 + 50z_4 + 200z_5, \\ & z_1 + 5z_2 + 30z_3 + 120z_4 + 360z_5 \leq 396 \end{aligned}$$

with upper bounds $u_1 = 4$, $u_2 = 68$, $u_3 = 2$, $u_4 = 1$, and $u_5 = 1$ on the variables z_i , $i = 1, \dots, 5$.

The following shows that for a given instance of the sequential knapsack problem (SKP), the partition of items into maximal blocks is unique. Let $V_k = \{i_1, \dots, i_l\}$ with $i_1 < i_2 < \dots < i_l$ be the set of all items in N with gain per unit equal to g_k , and consider the subset $\{i_1, \dots, i_t\}$ of V_k where $t = l$ if $a_{i_{j+1}} \leq \sum_{v=1}^j (s_{i_v} a_{i_v}) + a_{i_1}$ for all $j = 1, \dots, l-1$, and $t = \min\{j \in \{1, \dots, l-1\} : a_{i_{j+1}} > \sum_{v=1}^j (s_{i_v} a_{i_v}) + a_{i_1}\}$ otherwise. By construction and Definition 2.1, $\{i_1, \dots, i_t\}$ is a block. We now consider any partition of V_k into maximal blocks and any block B of this partition containing at least one item in $\{i_1, \dots, i_t\}$. Such a block B satisfies $i_r \in B \subseteq \{i_r, \dots, i_l\}$ for some $r \in \{1, \dots, t\}$ (i.e., i_r is the first element of block B). By definition of t , $a_{i_r} \leq \sum_{v=1}^{r-1} (s_{i_v} a_{i_v}) + a_{i_1}$ and no item i_j with $t < j \leq l$ can belong to B because $a_{i_j} \geq a_{i_{t+1}} > \sum_{v=1}^t (s_{i_v} a_{i_v}) + a_{i_1} \geq (\sum_{v=1}^t (s_{i_v} a_{i_v}) + a_{i_1}) + (a_{i_r} - \sum_{v=1}^{r-1} (s_{i_v} a_{i_v}) - a_{i_1}) = \sum_{v=r}^t (s_{i_v} a_{i_v}) + a_{i_r}$. Hence $B \subseteq \{i_r, \dots, i_t\}$, and thus all the maximal blocks containing at least one element in $\{i_1, \dots, i_t\}$ form a partition of $\{i_1, \dots, i_t\}$. As $\{i_1, \dots, i_t\}$ is a block, the unique maximal block containing i_1 is $B = B_1^k = \{i_1, \dots, i_t\}$. By removing B_1^k from V_k and iteratively using the same argument, the unique partition of V_k into maximal blocks $B_1^k, \dots, B_{n_k}^k$ can be constructed easily. Each block B_j^k , $j = 1, \dots, n_k$, of this partition corresponds to a subset of consecutive elements of V_k (where elements of V_k are ordered in increasing item number); i.e., $B_j^k = \{i_{s(j)}, \dots, i_{e(j)}\}$ for $j = 1, \dots, n_k$, $s(1) = 1$, $e(n_k) = l$, $e(j-1) + 1 = s(j)$ for $j = 2, \dots, n_k$. This argument applies to all sets of items V_k with $k \in \{1, \dots, v\}$.

From these discussions follows that, if we define $MSKP$ using the unique partition into maximal blocks, a vector z is feasible for $MSKP$ if and only if the associated

vectors x are feasible for SKP . As each maximal block contains items in N with the same gain per unit, we obtain in addition that a vector $x \in \mathbb{Z}^n$ is optimal with respect to SKP if and only if the associated vector $z \in \mathbb{Z}^m$ is optimal with respect to $MSKP$, and, conversely, a vector $z \in \mathbb{Z}^m$ is optimal with respect to $MSKP$ if and only if all of its associated vectors $x \in \mathbb{Z}^n$ are optimal solutions to SKP .

We assume from now on that $MSKP$ is defined using maximal blocks. Moreover, the items of an instance of $MSKP$ are ordered such that $f_1 \leq f_2 \leq \dots \leq f_m$, and in the case $f_w = f_{w+1}$, then $c_w > c_{w+1}$. This assumption can be made without loss of generality because the union of two blocks having the same weight is always a block. So, the only case in which two different maximal blocks may have the same weight is when their objective coefficients (or gains per unit) are different. From the construction of the unique partition into maximal blocks we conclude that for the transformed problem $MSKP$ the property

$$f_w > \sum_{i \in \{1, \dots, w-1\}: c_i/f_i = c_w/f_w} f_i u_i \quad \text{for } w = 2, \dots, m$$

always holds. This can be seen by showing that if B_w is the first block for which the above property does not hold, then $B_w \cup B_v$ is a block, where $v = \max\{i \in \{1, \dots, w-1\} : c_i/f_i = c_w/f_w\}$, and therefore the blocks are not maximal. This inequality is used in the next section to derive a decomposition scheme for all optimal solutions of $MSKP$.

3. Decomposition of optimal solutions. In this section we characterize the optimal solutions of problem $MSKP$ obtained by the maximal block transformation of SKP . The following additional notations are needed.

For $F \in \mathbb{N}$ and $j \in M = \{1, \dots, m\}$, we denote by $\mathcal{P}_j(F)$ the polyhedron

$$\mathcal{P}_j(F) = \left\{ z \in \mathbb{Z}^j : \sum_{i=1}^j f_i z_i \leq F, 0 \leq z_i \leq u_i \text{ for all } i = 1, \dots, j \right\}.$$

The optimization problem $OP_j(F)$ is the integer program

$$(OP_j(F)) \quad \max \sum_{i=1}^j c_i z_i : z \in \mathcal{P}_j(F).$$

By $O_j(F)$ we denote the set of all optimal solutions to $OP_j(F)$. For $j \in M$, we define Δ_j as the set of all items before j whose gain per unit is strictly better than that of j . In formulas,

$$\Delta_j = \left\{ l \in \{1, \dots, j-1\} : \frac{c_l}{f_l} > \frac{c_j}{f_j} \right\}.$$

With the parameters F and j we associate a decomposition tree in which the paths from the root node to the leaves contain (and possibly properly contain) all the optimal solutions of $OP_j(F)$. The key to this result is the next lemma, which shows that for every optimal solution $z \in O_j(F)$ the variable z_j can attain at most two different values.

LEMMA 3.1. *For the optimization problem $OP_j(F)$ with positive cost coefficients, the following implication is true:*

$$\begin{aligned} \text{if } z \in O_j(F), \quad & \text{then } z_j \geq \min \left\{ u_j; \left\lfloor \frac{(F - f(\Delta_j))^+}{f_j} \right\rfloor \right\} \\ \text{and } z_j \leq \min \left\{ u_j; \left\lceil \frac{(F - f(\Delta_j))^+}{f_j} \right\rceil \right\}. \end{aligned}$$

Proof. We prove this result by contradiction using standard exchange arguments. Several cases are distinguished.

(i) When $f(\Delta_j) \geq F$, the lemma states that $z_j = 0$ for all $z \in O_j(F)$. By contradiction, suppose that there exists $z \in O_j(F)$ with $z_j > 0$. As $\sum_{l \in \Delta_j} f_l z_l + f_j z_j \leq F \leq f(\Delta_j)$ and $z_j > 0$, we have $\sum_{l \in \Delta_j} f_l (u_l - z_l) \geq f_j z_j > 0$. By the divisibility of the weights, there exist integers $\lambda_l \in \{0, \dots, u_l - z_l\}$ for all $l \in \Delta_j$ such that $\sum_{l \in \Delta_j} f_l \lambda_l = f_j z_j$. We now define a solution z' with $z'_l = z_l$ for $l \in \{1, \dots, j-1\} \setminus \Delta_j$, $z'_l = z_l + \lambda_l \leq u_l$ for $l \in \Delta_j$, and $z'_j = 0$. Then, $z' \in \mathcal{P}_j(F)$ because $\sum_{i=1}^j f_i z_i = \sum_{i=1}^j f_i z'_i$ and the solution z' has strictly better objective value than z by definition of Δ_j . This contradicts the optimality of z .

(ii) When $f(\Delta_j) + f_j u_j \leq F$, we obtain $u_j \leq \left\lfloor \frac{(F - f(\Delta_j))^+}{f_j} \right\rfloor$ because u_j is integral. In this case the lemma states that $z_j = u_j$ for all $z \in O_j(F)$. By contradiction, suppose that there exists $z \in O_j(F)$ with $z_j < u_j$ and set $\delta = \sum_{i \in \{1, \dots, j-1\} \setminus \Delta_j} f_i z_i$.

If $\delta < f_j(u_j - z_j)$, the new solution z' with $z'_i = 0$ for $i \in \{1, \dots, j-1\} \setminus \Delta_j$, $z'_i = z_i$ for $i \in \Delta_j$, and $z'_j = u_j$ belongs to $\mathcal{P}_j(F)$ and has strictly better objective value than z because $\frac{c_j}{f_j} \geq \frac{c_i}{f_i}$ for all $i \in \{1, \dots, j-1\} \setminus \Delta_j$, which is a contradiction.

Hence, we can assume that $\delta \geq f_j(u_j - z_j)$. By the divisibility of the weights, there exist integers $\lambda_l \in \{0, \dots, z_l\}$ for all $l \in \{1, \dots, j-1\} \setminus \Delta_j$ with $\sum_{l \in \{1, \dots, j-1\} \setminus \Delta_j} f_l \lambda_l = f_j(u_j - z_j)$. The new solution z' with $z'_l = z_l - \lambda_l$ for $l \in \{1, \dots, j-1\} \setminus \Delta_j$, $z'_l = z_l$ for $l \in \Delta_j$, and $z'_j = u_j$ belongs to $\mathcal{P}_j(F)$.

Let $W = \{i \in \{1, \dots, j-1\} \setminus \Delta_j : \lambda_i > 0\}$. As $\sum_{i \in W} f_i u_i \geq \sum_{i \in W} f_i \lambda_i = f_j(u_j - z_j) \geq f_j > \sum_{i=1, \frac{c_i}{f_i} = \frac{c_j}{f_j}}^{j-1} f_i u_i$ (where the last inequality holds by the partition into maximal blocks), there exists $i \in W$ with $\frac{c_i}{f_i} < \frac{c_j}{f_j}$. Then the solution z' has strictly better objective value than z , again a contradiction.

In the remaining cases we have that $F - f_j u_j < f(\Delta_j) < F$, and the lemma states that $z_j \geq \left\lfloor \frac{F - f(\Delta_j)}{f_j} \right\rfloor$ and $z_j \leq \left\lceil \frac{F - f(\Delta_j)}{f_j} \right\rceil$.

(iii) Let $F - f_j u_j < f(\Delta_j) < F$, and suppose that there exists $z \in O_j(F)$ with $z_j < \left\lfloor \frac{F - f(\Delta_j)}{f_j} \right\rfloor$. As $f(\Delta_j) + \left\lfloor \frac{F - f(\Delta_j)}{f_j} \right\rfloor f_j \leq F$, a similar argument to that in case (ii) shows that there exists a solution $z' \in \mathcal{P}_j(F)$ with $z'_i = z_i$ for $i \in \Delta_j$, $z'_j = \left\lfloor \frac{F - f(\Delta_j)}{f_j} \right\rfloor > z_j$, $\sum_{i \in \{1, \dots, j-1\} \setminus \Delta_j} f_i z'_i = \left[\sum_{i \in \{1, \dots, j-1\} \setminus \Delta_j} f_i z_i - f_j(z'_j - z_j) \right]^+$, and with a strictly better objective value than z . This is a contradiction.

(iv) Let $F - f_j u_j < f(\Delta_j) < F$, and suppose that there exists $z \in O_j(F)$ with $z_j > \left\lceil \frac{F - f(\Delta_j)}{f_j} \right\rceil$. As $\sum_{l \in \Delta_j} f_l z_l + f_j z_j \leq F \leq f(\Delta_j) + \left\lceil \frac{F - f(\Delta_j)}{f_j} \right\rceil f_j$, a similar argument to that in case (i) shows that there exists a solution $z' \in \mathcal{P}_j(F)$ with $z'_i = z_i$ for all $i \in \{1, \dots, j-1\} \setminus \Delta_j$, $z'_j = \left\lceil \frac{F - f(\Delta_j)}{f_j} \right\rceil < z_j$, $\sum_{i \in \Delta_j} f_i z'_i = \left[\sum_{i \in \Delta_j} f_i z_i + f_j(z_j - z'_j) \right]$, and with a strictly better objective

value than z , which is a contradiction. $\square$

Lemma 3.1 can be applied inductively to build a binary decomposition tree containing all potential optimal solutions in $O_j(F)$. We illustrate this on our initial example.

Example 3.2 (2.3 continued). The modified sequential knapsack problem $OP_5(396)$ is the integer program

$$\begin{aligned} \max \quad & z_1 + 3z_2 + 6z_3 + 50z_4 + 200z_5, \\ & z_1 + 5z_2 + 30z_3 + 120z_4 + 360z_5 \leq 396 \end{aligned}$$

with upper bounds $u_1 = 4$, $u_2 = 68$, $u_3 = 2$, $u_4 = 1$, and $u_5 = 1$.

We have $\frac{c_1}{f_1} = 1$, $\frac{c_2}{f_2} = 0.6$, $\frac{c_3}{f_3} = 0.2$, $\frac{c_4}{f_4} = 0.42$, $\frac{c_5}{f_5} = 0.56$, and hence $\Delta_1 = \emptyset$, $\Delta_2 = \{1\}$, $f(\Delta_2) = 4$, $\Delta_3 = \Delta_4 = \Delta_5 = \{1, 2\}$, $f(\Delta_3) = f(\Delta_4) = f(\Delta_5) = 344$.

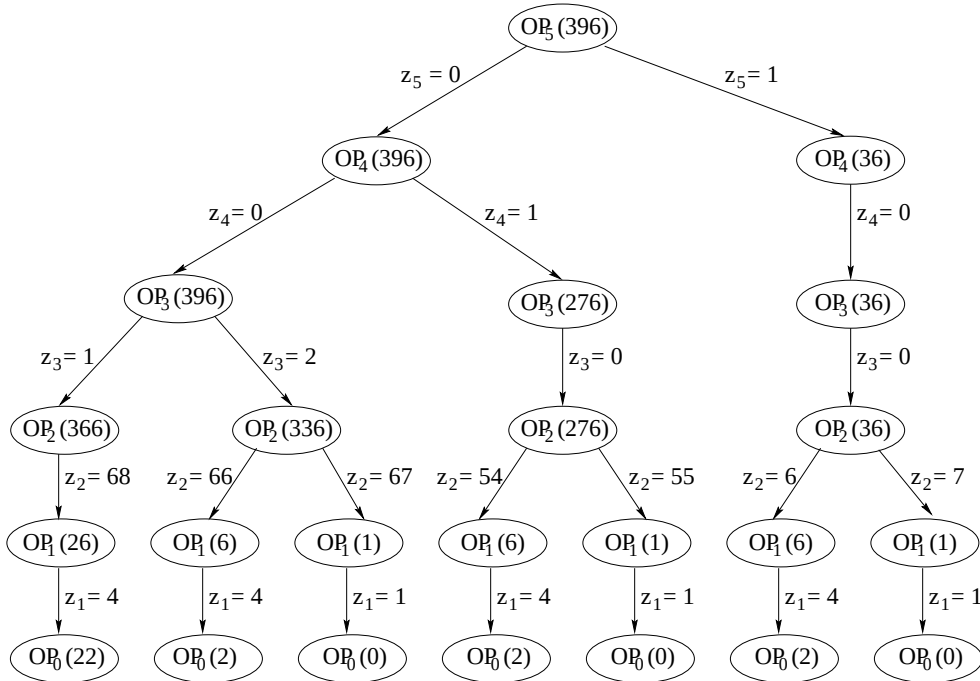


FIG. 1. *Decomposition of optimal solutions for Example 2.3.*

Figure 1 illustrates the decomposition tree that we obtain from a repeated application of Lemma 3.1. The node labels identify the problems $OP_j(F)$ to be solved and the value of z_j is fixed on the corresponding branches. For example, Lemma 3.1 applied to $OP_5(396)$ yields $z_5 = 0$ or $z_5 = 1$. If $z_5 = 0$ we are left with problem $OP_4(396)$, and if $z_5 = 1$ we are left with problem $OP_4(36)$. Potential optimal solutions of problems $OP_4(396)$ and $OP_4(36)$ are further decomposed using Lemma 3.1.

The set $S_5(396)$ of potential optimal solutions to $OP_5(396)$ is defined by all the paths from the leaves to the root node in the decomposition tree; that is,

$$S_5(396) = \left\{ (4, 68, 1, 0, 0), (4, 66, 2, 0, 0), (1, 67, 2, 0, 0), \right. \\ \left. (4, 54, 0, 1, 0), (1, 55, 0, 1, 0), (4, 6, 0, 0, 1), \right. \\ \left. (1, 7, 0, 0, 1) \right\}$$

and, by Lemma 3.1, $O_5(396) \subseteq S_5(396)$.

For a given problem $OP_j(F)$ and its associated decomposition tree, we define in the next section valid inequalities that are satisfied at equality by all solutions in this decomposition tree and thus by all optimal solutions in $O_j(F)$.

4. The convex hull of all solutions to the sequential knapsack problem.

Let SKP be an instance of the sequential knapsack problem and suppose that $\mathcal{C}$ is a class of valid inequalities for $\mathcal{P}_{SKP}$. The technique that we use in order to show that $\mathcal{C}$ describes $\mathcal{P}_{SKP}$ is due to Lovász [5]: For every objective function γ , we prove that the set of optimal solutions to SKP is included in the face induced by some inequality in $\mathcal{C}$. This suffices to show that $\mathcal{C}$ describes $\mathcal{P}_{SKP}$.

We first consider the case where all objective function coefficients are positive. As outlined in section 2, we partition N into maximal blocks $B_1, \dots, B_m$ and construct the modified sequential knapsack problem ($MSKP$). Associated with the transformed problem ($MSKP$), we use the notation $\mathcal{P}_j(F)$, $OP_j(F)$, $O_j(F)$, Δ_j , and $f(\Delta_j)$, introduced in section 3.

For every knapsack capacity $F \in \mathbb{N}$ and for every $j \in M = \{1, \dots, m\}$, we now define an inequality $I_j(F)$ satisfying the three conditions listed below:

- (1) If $F = F' \pmod{f_j}$, then $I_j(F)$ and $I_j(F')$ have the same left-hand sides.
- (2) $I_j(F)$ is a valid inequality for $\mathcal{P}_j(F)$.
- (3) $O_j(F)$ is contained in the face induced by $I_j(F)$.

The inequalities $I_j(F)$ are defined inductively on j .

If $j = 1$, for any integer F we define $I_1(F)$ as the inequality $z_1 \leq \min\{F, u_1\}$. This inequality clearly satisfies the properties (1), (2), (3).

The inductive step $j - 1 \rightarrow j$. Let $F \in \mathbb{N}$ be given and set $r := F \pmod{f_j}$. By assumption of the induction there exists an inequality $I_{j-1}(F)$ that satisfies the properties (1), (2), (3). Let $I_{j-1}(F)$ be written as $\sum_{i=1}^{j-1} d_i z_i \leq g_{j-1}(F)$. (Note that by property (1), and as f_{j-1} divides f_j , all inequalities $I_{j-1}(F')$ with $F' = F \pmod{f_j}$ will have the same left-hand side as $I_{j-1}(F)$.)

With j and r we associate an integer $F_j(r)$ defined as

$$F_j(r) = \begin{cases} r & \text{if } f(\Delta_j) < r, \\ \max\{G \in \mathbb{N} \mid G \leq f(\Delta_j) \text{ and } G \pmod{f_j} = r\} & \text{if } f(\Delta_j) \geq r. \end{cases}$$

We now define the inequality $I_j(F)$ for $j \geq 2$. The left-hand side is of the form $\sum_{i=1}^j d_i z_i$, where the integers $d_1, \dots, d_{j-1}$ are the coefficients of the left-hand side of $I_{j-1}(F)$ and d_j is defined by

$$d_j := g_{j-1}(F_j(r) + f_j) - g_{j-1}(F_j(r)).$$

In order to define the right-hand side of $I_j(F)$ (which we denote by $g_j(F)$) we need to distinguish several cases.

Compute the integer s by $s := (F - F_j(r))/f_j$, and write $F = F_j(r) + sf_j$. We define

$$g_j(F) := \begin{cases} g_{j-1}(F_j(r)) + sd_j & \text{if } 0 \leq s \leq u_j, \\ g_{j-1}(F) & \text{if } s < 0, \\ g_{j-1}(F - f_j u_j) + u_j d_j & \text{if } s > u_j. \end{cases}$$

Note that $g_1(F) := \min\{F, u_1\}$ and that $g_j(F)$ always satisfies the inequality $g_j(F) \leq \sum_{i=1}^j u_i d_i$. We will now illustrate our (inductive) construction on the initial example. Then we will derive properties of the function $g_j(F)$ that we use afterwards in order to show that the inequality $I_j(F)$ that reads $\sum_{i=1}^j d_i z_i \leq g_j(F)$ satisfies the conditions (1), (2), (3).

Example 4.1 (2.3 continued).

$$\begin{aligned} \max \quad & z_1 + 3z_2 + 6z_3 + 50z_4 + 200z_5, \\ & z_1 + 5z_2 + 30z_3 + 120z_4 + 360z_5 \leq 396 \end{aligned}$$

with upper bounds $u_1 = 4$, $u_2 = 68$, $u_3 = 2$, $u_4 = 1$, and $u_5 = 1$ on the variables z_i , $i = 1, \dots, 5$. Note that the construction of the inequalities is defined for any value of F , although we want to derive in this example precisely one inequality, $I_5(396)$. It is then not necessary to establish $I_j(F)$ for all values of F , but certain values of F can be excluded in advance. The node labels in Figure 1 represent the subproblems we have to solve in order to obtain an optimal solution for the original problem $OP_5(396)$. From these subproblems we can derive the relevant values for F and r in order to construct $I_5(396)$.

$$\begin{aligned} I_1(F) : \quad & z_1 \leq g_1(F) = \min\{F, 4\} \\ I_2(F) \quad & \text{with } r = F \bmod f_2 = 396 \bmod 5 = 1, \\ & \text{and } \Delta_2 = \{1\}, f(\Delta_2) = 4, F_2(r) = 1, F_2(r) + f_2 = 6. \\ I_2(1) : \quad & z_1 + (g_1(6) - g_1(1))z_2 \leq g_2(1) := g_1(1) + 0(g_1(6) - g_1(1)) \\ & z_1 + 3z_2 \leq 1 \\ I_2(6) : \quad & z_1 + (g_1(6) - g_1(1))z_2 \leq g_2(6) := g_1(1) + 1(g_1(6) - g_1(1)) \\ & z_1 + 3z_2 \leq 4 \\ I_2(1 + 5s) : \quad & z_1 + 3z_2 \leq 1 + 3s \quad \text{for } 1 \leq s \leq 68 \\ I_2(1 + 5s) : \quad & z_1 + 3z_2 \leq g_1(1 + 5(s - 68)) + 68 * 3 = 208 \quad \text{for } s > 68 \\ I_3(F) \quad & \text{with } r = F \bmod f_3 = 396 \bmod 30 = 6, \\ & \text{and } \Delta_3 = \{1, 2\}, f(\Delta_3) = 344, F_3(r) = 336, F_3(r) + f_3 = 366. \\ I_3(336) : \quad & z_1 + 3z_2 + (g_2(366) - g_2(336))z_3 \leq g_3(336) := g_2(336) \\ & z_1 + 3z_2 + 6z_3 \leq 202 \\ I_3(366) : \quad & z_1 + 3z_2 + 6z_3 \leq g_3(366) := g_2(336) + 1(g_2(366) - g_2(336)) \\ & z_1 + 3z_2 + 6z_3 \leq 208 \\ I_3(396) : \quad & z_1 + 3z_2 + 6z_3 \leq 214 \\ I_4(F) \quad & \text{with } r = F \bmod f_4 = 396 \bmod 120 = 36, \\ & \text{and } \Delta_4 = \{1, 2\}, f(\Delta_4) = 344, F_4(r) = 276, F_4(r) + f_4 = 396. \\ I_4(276) : \quad & z_1 + 3z_2 + 6z_3 + (g_3(396) - g_3(276))z_4 \leq \\ & g_4(276) := g_3(276) = g_2(276) \\ & z_1 + 3z_2 + 6z_3 + 48z_4 \leq 166 \\ I_4(396) : \quad & z_1 + 3z_2 + 6z_3 + 48z_4 \leq \\ & g_4(396) := g_3(276) + 1(g_3(396) - g_3(276)) \\ & z_1 + 3z_2 + 6z_3 + 48z_4 \leq 214 \\ I_5(F) \quad & \text{with } r = F \bmod f_5 = 396 \bmod 360 = 36, \\ & \text{and } \Delta_5 = \{1, 2\}, f(\Delta_5) = 344, F_5(r) = 36, F_5(r) + f_5 = 396. \\ I_5(36) : \quad & z_1 + 3z_2 + 6z_3 + 48z_4 + (g_4(396) - g_4(36))z_5 \leq g_5(36) := g_4(36) \\ & z_1 + 3z_2 + 6z_3 + 48z_4 + 192z_5 \leq 22 \\ I_5(396) : \quad & z_1 + 3z_2 + 6z_3 + 48z_4 + 192z_5 \leq \\ & g_5(396) := g_3(36) + g_4(396) - g_4(36) \\ & z_1 + 3z_2 + 6z_3 + 48z_4 + 192z_5 \leq 214 \end{aligned}$$

It is easy to check that the inequality $I_5(396)$ is satisfied at equality by all solutions

in $S_5(396)$ (defined by all the paths from leaves to the root node in Figure 1) and thus by all optimal solutions in $O_5(396) \subseteq S_5(396)$.

LEMMA 4.2. *For any $j = 1, \dots, n$ and any natural number F , the function $g_j(F + sf_j)$ is a concave nondecreasing function of s (where the function is defined over all integers s such that $F + sf_j \geq 0$).*

Proof. Since $g_1(F) = \min\{F, u_1\}$, the statement is true for $j = 1$. We assume inductively that it is true for $j - 1$. To prove the statement for j and any F , it suffices to show that

$$G_j(r, s) := g_j(F_j(r) + [s + 1]f_j) - g_j(F_j(r) + sf_j)$$

is a nonnegative nonincreasing function of s for all r such that $0 \leq r \leq f_j - 1$. We obtain

$$G_j(r, s) = \begin{cases} g_{j-1}(F_j(r) + [s + 1]f_j) - g_{j-1}(F_j(r) + sf_j) & \text{if } s < 0, \\ d_j = g_{j-1}(F_j(r) + f_j) - g_{j-1}(F_j(r)) & \text{if } 0 \leq s < u_j, \\ g_{j-1}(F_j(r) + [s - u_j + 1]f_j) - g_{j-1}(F_j(r) + [s - u_j]f_j) & \\ \text{otherwise.} \end{cases}$$

By induction, $g_{j-1}(H + hf_j) = g_{j-1}(H + [hf_j/f_{j-1}]f_{j-1})$ is a concave nondecreasing function of h , for any integer H . This implies the statement. $\square$

The following two properties are an immediate consequence of the fact that $g_j(F + sf_j)$ is a concave nondecreasing function of s .

LEMMA 4.3. *Let F and G be natural numbers such that $F \leq G$ and $F \bmod f_j = r = G \bmod f_j$. We have*

- (i) $g_j(G) + \sigma(g_j(F + f_j) - g_j(F)) \geq g_j(G + \sigma f_j)$ for every $\sigma \in \mathbb{N}$.
- (ii) $g_j(F - \sigma f_j) + \sigma(g_j(G) - g_j(G - f_j)) \leq g_j(F)$ for every $\sigma \in \mathbb{N}$ such that $F - \sigma f_j \geq 0$.

Using Lemmas 4.2 and 4.3 we are now able to prove the following theorem.

THEOREM 4.4. *Let a modified sequential knapsack problem with positive objective function obtained from the maximal block transformation be given. Let $j \in \{2, \dots, m\}$. If, for all $F \in \mathbb{N}$, the inequality $I_{j-1}(F)$ satisfies conditions (1), (2), and (3) (where j is replaced $j - 1$), then, for all $F \in \mathbb{N}$, the inequality $I_j(F)$ satisfies (1), (2), and (3).*

Proof. We write $I_j(F)$ as $\sum_{i=1}^j d_i z_i \leq g_j(F)$.

(a) Let F and F' be two natural numbers satisfying $F \bmod f_j = r = F' \bmod f_j$. As Δ_j is independent of F , and $F_j(r)$ is uniquely defined by Δ_j and r , the coefficient d_j attains the same value in the inequalities $I_j(F)$ and $I_j(F')$. As $F = F' \bmod f_{j-1}$, the left-hand sides of the two inequalities $I_j(F)$ and $I_j(F')$ coincide.

(b) The following shows that the inequality $I_j(F)$ is valid for the polytope $\mathcal{P}_j(F)$. Let $z \in \mathcal{P}_j(F)$ be a feasible point, then

$$\sum_{i=1}^j d_i z_i \leq g_{j-1}(F - z_j f_j) + z_j d_j$$

because, by condition (2), $\sum_{i=1}^{j-1} d_i z_i \leq g_{j-1}(G)$ is a valid inequality for $\mathcal{P}_{j-1}(G)$ for all values of G with $G \bmod f_{j-1} = F \bmod f_{j-1}$ and, by the divisibility property, $(F - z_j f_j) \bmod f_{j-1} = F \bmod f_{j-1}$. Again, we write $r = F \bmod f_j$, $F = F_j(r) + sf_j$, and distinguish several cases.

If $F \leq F_j(r)$, we have $s \leq 0$. If $z_j = 0$, it follows from the definition of $g_j(F) = g_{j-1}(F)$ that the inequality is valid. Suppose that $z_j > 0$. Then $\sum_{i=1}^j d_i z_i \leq g_{j-1}(F -$

$z_j f_j + z_j d_j = g_{j-1}(F - z_j f_j) + z_j(g_{j-1}(F_j(r) + f_j) - g_{j-1}(F_j(r))) \leq g_{j-1}(F) = g_j(F)$, where the last inequality follows from Lemma 4.3(ii). The statement follows.

If $F > F_j(r) + f_j u_j$, we have $s > u_j$. If $z_j = u_j$, it follows from the definition of $g_j(F) = g_{j-1}(F - f_j u_j) + d_j u_j$ that the inequality is valid. Suppose that $z_j < u_j$. By applying Lemma 4.3(i), we obtain $\sum_{i=1}^j d_i z_i \leq g_{j-1}(F - z_j f_j) + z_j d_j = g_{j-1}(F - z_j f_j) + z_j(g_{j-1}(F_j(r) + f_j) - g_{j-1}(F_j(r))) \leq g_{j-1}(F - u_j f_j) + (u_j - z_j)(g_{j-1}(F_j(r) + f_j) - g_{j-1}(F_j(r))) + z_j(g_{j-1}(F_j(r) + f_j) - g_{j-1}(F_j(r))) = g_{j-1}(F - u_j f_j) + u_j d_j = g_j(F)$.

What remains is the case where $F_j(r) < F \leq F_j(r) + u_j f_j$. Then $1 \leq s \leq u_j$ holds and we obtain

$$(4) \quad \sum_{i=1}^j d_i z_i \leq g_{j-1}(F - z_j f_j) + z_j d_j = g_{j-1}(F_j(r) - (z_j - s)f_j) + (z_j - s)d_j + s d_j.$$

If $z_j = s$, then the inequality is satisfied because $g_j(F) = g_{j-1}(F_j(r)) + s d_j$. If $z_j > s$, then Lemma 4.3(ii) implies that $g_{j-1}(F_j(r) - (z_j - s)f_j) + (z_j - s)d_j \leq g_{j-1}(F_j(r))$ and together with (4) we have that $I_j(F)$ is valid. Finally, if $z_j < s$, Lemma 4.3(i) implies that $g_{j-1}(F_j(r) - (z_j - s)f_j) + (z_j - s)d_j = g_{j-1}(F_j(r) + (s - z_j)f_j) - (s - z_j)d_j \leq g_{j-1}(F_j(r))$, which again shows that the inequality $I_j(F)$ is valid.

(c) It remains to be shown that the set of optimal solution $O_j(F)$ is contained in the face induced by the inequality $I_j(F)$.

Let $z \in O_j(F)$, define $r = F \bmod f_j$, and write F in the form $F = F_j(r) + s f_j$. By Lemma 3.1 and the definition of $F_j(r)$, we have $(F - f(\Delta_j))/f_j = s + (F_j(r) - f(\Delta_j))/f_j$. If $r \leq f(\Delta_j)$, then $F_j(r)$ is the unique number such that $F_j(r) \leq f(\Delta_j) < F_j(r) + f_j$ and $F_j(r) \bmod f_j = r$. In this case Lemma 3.1 implies that

$$z_j = \min\{u_j, (s - 1)^+\} \text{ or } z_j = \min\{u_j, s^+\}.$$

If $r > f(\Delta_j)$, then $F_j(r) = r < f_j$, $s \geq 0$, and Lemma 3.1 yields

$$z_j = \min\{u_j, s\} \text{ or } z_j = \min\{u_j, s + 1\}.$$

In this case, $z_j = s + 1$ is impossible because $F - (s + 1)f_j = F_j(r) - f_j < 0$. Hence, $r > f(\Delta_j)$ implies that $z_j = \min\{u_j, s\}$. Summarizing all cases yields $z_j = \min\{u_j, (s - 1)^+\}$ or $z_j = \min\{u_j, s^+\}$.

In the case $s \leq 0$, i.e., $F \leq F_j(r)$, we have $z_j = 0$ in every optimum solution. Therefore, by assumption of the induction, every optimal solution to problem $OP_j(F)$ is contained in the face $\sum_{i=1}^{j-1} d_i z_i = g_{j-1}(F)$. Since $g_j(F) = g_{j-1}(F)$ in this case, the claim follows.

In the case $s \geq u_j + 1$, i.e., $F > F_j(r) + u_j f_j$, every element in the set $O_j(F)$ satisfies $z_j = u_j$. By the induction assumption, every optimum solution z to problem $OP_j(F)$ satisfies $\sum_{i=1}^{j-1} d_i z_i = g_{j-1}(F - f_j u_j)$, and as $z_j = u_j$, we obtain $\sum_{i=1}^j d_i z_i = g_{j-1}(F - f_j u_j) + d_j u_j = g_j(F)$. This proves the claim in this case.

Finally, we have $1 \leq s \leq u_j$. Then every optimal solution z of $OP_j(F)$ satisfies either $z_j = s$ or $z_j = s - 1$. From the assumption of the induction one may derive (a) $\sum_{i=1}^{j-1} d_i z_i = g_{j-1}(F - s f_j) = g_{j-1}(F_j(r))$ if $z_j = s$ and (b) $\sum_{i=1}^{j-1} d_i z_i = g_{j-1}(F - (s - 1)f_j) = g_{j-1}(F_j(r) + f_j)$ if $z_j = s - 1$. In case (a) this yields $\sum_{i=1}^j d_i z_i = g_{j-1}(F_j(r)) + s d_j = g_j(F)$. In case (b) we obtain $\sum_{i=1}^j d_i z_i = g_{j-1}(F_j(r) + f_j) + (s - 1)d_j = g_{j-1}(F_j(r)) + (g_{j-1}(F_j(r) + f_j) - g_{j-1}(F_j(r))) + (s - 1)d_j = g_{j-1}(F_j(r)) + s d_j = g_j(F)$. This shows that in both cases the inequality $I_j(F)$ is satisfied as an equation by all optimal points. $\square$

Let us now present the final theorem describing $\mathcal{P}_{SKP}$ as a system of inequalities. Let $W \subseteq N$ be a subset of items in N , let $\mathcal{B} = \{B_1, \dots, B_m\}$ be a partition of W into blocks, and let π be a permutation of $\{1, \dots, m\}$. Let $f'_i := \min_{l \in B_i} \{a_l\}$ be the weight of block B_i , let $u_i := \sum_{l \in B_i} \frac{a_l s_l}{f'_i}$ be the multiplicity of block B_i , and assume that $f'_1 \leq \dots \leq f'_m$. We set $f_j = f'_j / f'_1$ and $\Delta_j = \{i \in \{1, \dots, j-1\} : \pi(i) < \pi(j)\}$, $j = 1, \dots, m$.

Note that the inequalities constructed in this section do not depend directly on the objective function coefficients, but only on the ordering of blocks defined by Δ_j , $j = 1, \dots, m$. So, different positive objective functions with the same decomposition into maximal blocks and the same ordering of blocks (defined by Δ_j , $j = 1, \dots, m$) would give the same inequalities $I_j(F)$ for all values of $j = 1, \dots, m$ and F .

Denote by $\mathcal{P}_m^{\mathcal{B}(W)}(\lfloor a_0 / f'_1 \rfloor)$ the modified knapsack polytope defined with the block partition $\mathcal{B}$ of W , weights $f_1, \dots, f_m$, multiplicities $u_1, \dots, u_m$, and knapsack capacity $\lfloor a_0 / f'_1 \rfloor$. That is

$$\mathcal{P}_m^{\mathcal{B}(W)}(\lfloor a_0 / f'_1 \rfloor) = \text{conv} \left\{ z \in \mathbb{Z}^m : \sum_{i=1}^m f_i z_i \leq \lfloor a_0 / f'_1 \rfloor, 0 \leq z_i \leq u_i \text{ for } i = 1, \dots, m \right\}.$$

If the inequality $I_m(\lfloor a_0 / f'_1 \rfloor)$, written as $\sum_{j=1}^m d_j z_j \leq g_m(\lfloor a_0 / f'_1 \rfloor)$, denotes the valid inequality developed in this section for $\mathcal{P}_m^{\mathcal{B}(W)}(\lfloor a_0 / f'_1 \rfloor)$ using the sets Δ_j induced by the permutation π , then the inequality $K(W, \mathcal{B}, \pi)$ is defined as

$$\sum_{j=1}^m \sum_{i \in B_j} d_j \left(\frac{a_i}{f_j f'_1} \right) x_i \leq g_m(\lfloor a_0 / f'_1 \rfloor).$$

THEOREM 4.5. *The following system of inequalities describes the polytope $\mathcal{P}_{SKP}$:*

$$\begin{aligned} 0 &\leq x_i && \text{for } i = 1, \dots, n, \\ K(W, \mathcal{B}, \pi) &&& \text{for all } W \subseteq N, \text{ all partitions } \mathcal{B} = \{B_1, \dots, B_m\} \text{ of} \\ &&& W \text{ into blocks, and all permutations } \pi \text{ of } \{1, \dots, m\}. \end{aligned}$$

Proof. We first show validity of the inequalities $K(W, \mathcal{B}, \pi)$. Given W , $\mathcal{B} = \{B_1, \dots, B_m\}$ and π . It is easy to check that there exists an objective function $\gamma \in \mathbb{R}^{|W|}$ for which $\mathcal{B}$ is the partition of W into maximal blocks and there exists a permutation π such that

$$\Delta_j = \left\{ i \in \{1, \dots, j-1\} : \frac{\gamma_i}{f_i} > \frac{\gamma_j}{f_j} \right\}.$$

Then the inequality $I_m(\lfloor a_0 / f'_1 \rfloor)$ is valid for the polytope $\mathcal{P}_m^{\mathcal{B}(W)}(\lfloor a_0 / f'_1 \rfloor)$ by Theorem 4.4. By the arguments on the transformation of valid inequalities for $\mathcal{P}_{MSKP}$ to valid inequalities for $\mathcal{P}_{SKP}$ (see section 2), the inequality $K(W, \mathcal{B}, \pi)$ is valid for the polytope

$$\text{conv} \left\{ x \in \mathbb{Z}^n : \sum_{i \in W} (a_i / f'_1) x_i \leq \lfloor a_0 / f'_1 \rfloor, 0 \leq x_i \leq s_i \text{ for } i \in W \right\}.$$

This polytope is a relaxation of $\mathcal{P}_{SKP}$ because f'_1 is the smallest weight among all items in W . As $K(W, \mathcal{B}, \pi)$ is valid for this relaxation of $\mathcal{P}_{SKP}$, it is certainly valid for $\mathcal{P}_{SKP}$.

For a given objective function $\gamma = (\gamma_1, \dots, \gamma_n)^T$, we now construct an inequality that is satisfied as an equation by all optimal solutions of SKP . If $\gamma_i < 0$ for some $i \in N$, then $x_i = 0$ for every optimal solution. Otherwise, $\gamma \geq 0$ and we set $W := \{i \in N : \gamma_i > 0\}$. Let $\mathcal{B} = \{B_1, \dots, B_m\}$ be the partition of W into maximal blocks, and let $MSKP$ denote the modified sequential knapsack problem of section 2 corresponding to item set W and objective function γ . From Theorem 4.4 we know that $I_m(\lfloor a_0/f'_1 \rfloor)$ is satisfied at equality by all optimal solutions of $MSKP$. By the arguments on the equivalence of optimal solutions between problems SKP and $MSKP$ (see section 2), $K(W, \mathcal{B}, \pi)$ is satisfied at equality by all optimal solutions of

$$\begin{aligned} \max \quad & \sum_{i \in W} \gamma_i x_i, \\ & \sum_{i \in W} (a_i/f'_1) x_i \leq \lfloor a_0/f'_1 \rfloor, \\ & x_i \in \{0, \dots, s_i\} \quad \text{for } i \in W. \end{aligned}$$

As $\gamma_i = 0$ for all $i \in N \setminus W$, a solution x of the original problem SKP , possibly with $x_i > 0$ for some $i \in N \setminus W$, can be optimal only if its components corresponding to items in W satisfy $K(W, \mathcal{B}, \pi)$ at equality. $\square$

5. Explicit inequalities. In the previous section we have inductively defined a class of inequalities that depends on the choice and ordering of the blocks. Can we find a more explicit or combinatorial formulation for those inequalities? This question is addressed now.

Let the following knapsack constraint define the set of feasible solutions of an instance of SKP ,

$$\sum_{i \in N_1} x_i + \sum_{i \in N_2} f_2 x_i + \dots + \sum_{i \in N_k} f_k x_i \leq F, \quad 0 \leq x \leq u, \quad x \text{ integer},$$

where u is the vector of upper bounds on the variables and $1 < f_2 < f_3 < \dots < f_k$.

One class of valid inequalities of $\mathcal{P}_{SKP}$ can be described as follows.

DEFINITION 5.1. *Let sets $S_1 \subseteq N_1, \dots, S_k \subseteq N_k$ such that $\sum_{j=1}^k f_j u(S_j) = F$ and sets $T_i \subseteq N_i \setminus S_i$, $i = 1, \dots, k$ satisfying $0 < u(T_1) < f_2$ and $T_k = N_k \setminus S_k$ be given. We set $b_1 := 1$ and, for $j \geq 2$, $b_j = \sum_{w=1}^{j-1} b_w u(T_w)$ if $\sum_{w=1}^{j-1} f_w u(T_w) < f_j$, and $b_j = (f_j/f_{j-1})b_{j-1}$, otherwise. With the vector b we associate the inequality*

$$(5) \quad \sum_{j=1}^k b_j \sum_{i \in S_j \cup T_j} x_i \leq \sum_{j=1}^k u(S_j) b_j.$$

An application of the inductive scheme of the previous section proves that inequality (5) is valid for $\mathcal{P}_{SKP}$. To this end we can define a modified sequential knapsack problem and, for every item i in this modified problem, we choose a set Δ_i such that the inequality constructed via our inductive scheme coincides with (5).

PROPOSITION 5.2. *Inequality (5) is valid for $\mathcal{P}_{SKP}$.*

Rather than giving a formal and technical proof of this result, we illustrate it with an example.

Example 5.3. Consider the instance of the sequential knapsack polytope $\mathcal{P}_{SKP}$ defined by

$$\begin{aligned} 1(x_1 + x_2 + x_3) + 5(x_4 + x_5 + x_6) + 10(x_7 + x_8 + x_9) \\ + 30(x_{10} + x_{11}) + 120(x_{12} + x_{13}) \leq 396 \end{aligned}$$

with $N_1 = \{1, 2, 3\}$, $N_2 = \{4, 5, 6\}$, $N_3 = \{7, 8, 9\}$, $N_4 = \{10, 11\}$, $N_5 = \{12, 13\}$, and upper bounds $u_1 = 1$, $u_2 = 3$, $u_3 = 5$, $u_4 = 1$, $u_5 = 3$, $u_6 = 5$, $u_7 = u_8 = u_9 = 3$, $u_{10} = u_{11} = 2$, $u_{12} = u_{13} = 3$ on the variables x_i , $i = 1, \dots, 13$.

The following sets S_i , T_i , and coefficients b_i for $i = 1, \dots, 5$ satisfy the conditions of Definition 5.1.

$$\begin{aligned}
S_1 &= \{1\}, & T_1 &= \{2\}, & b_1 &= 1, \\
S_2 &= \{4\}, & T_2 &= \emptyset, & b_2 &= \sum_{w=1}^1 b_w u(T_w) = 3 \quad \left(\sum_{w=1}^1 f_w u(T_w) = 3 < 5 = f_2 \right), \\
S_3 &= \{7\}, & T_3 &= \{8\}, & b_3 &= \sum_{w=1}^2 b_w u(T_w) = 3 \quad \left(\sum_{w=1}^2 f_w u(T_w) = 3 < 10 = f_3 \right), \\
S_4 &= \emptyset, & T_4 &= \{10\}, & b_4 &= (f_4/f_3)b_3 = 9 \quad \left(\sum_{w=1}^3 f_w u(T_w) = 33 \geq 30 = f_4 \right), \\
S_5 &= \{12\}, & T_5 &= \{13\}, & b_5 &= \sum_{w=1}^4 b_w u(T_w) = 30 \quad \left(\sum_{w=1}^4 f_w u(T_w) = 93 < 120 = f_5 \right), \\
&&&&& \text{and } \sum_{j=1}^5 b_j u(S_j) = 103, & \sum_{j=1}^5 f_j u(S_j) = 396.
\end{aligned}$$

The corresponding inequality (5)

$$1(x_1 + x_2) + 3(x_4) + 3(x_7 + x_8) + 9(x_{10}) + 30(x_{12} + x_{13}) \leq 103$$

defines a facet of this instance of $\text{conv}(\mathcal{P}_{SKP})$. Now we show how to obtain this inequality via our inductive scheme. First, we define the block transformation to $MSKP$. We start with the first block $B_1 = S_1 \cup T_1$. Then we consider each set $S_j \cup T_j$ from $j = 2$ to $j = k$. If $\sum_{w=1}^{j-1} f_w u(T_w) < f_j$, i.e., $b_j = \sum_{w=1}^{j-1} b_w u(T_w)$, we create a new block $B = S_j \cup T_j$. Otherwise, if $\sum_{w=1}^{j-1} f_w u(T_w) \geq f_j$, i.e., $b_j = (f_j/f_{j-1})b_{j-1}$, we add $S_j \cup T_j$ to the last block created. Let v be the number of blocks created.

In our example, this procedure creates $v = 4$ blocks $B_1 = S_1 \cup T_1$, $B_2 = S_2 \cup T_2$, $B_3 = S_3 \cup T_3 \cup S_4 \cup T_4$, $B_4 = S_5 \cup T_5$ of respective weights $1 = f_1$, $5 = f_2$, $10 = f_3$, $120 = f_5$ and multiplicities (upper bounds) $4 = u(B_1)/1$, $1 = u(B_2)/5$, $12 = u(B_3)/10$, $6 = u(B_4)/120$. The resulting $MSKP$ is defined as

$$1z_1 + 5z_2 + 10z_3 + 120z_4 \leq 396,$$

$$z_1 \leq 4, \quad z_2 \leq 1, \quad z_3 \leq 12, \quad z_4 \leq 6,$$

$$z_j \text{ integer for } j = 1, \dots, 4.$$

Here the variable z_i is associated with block B_i . In terms of the original variables, z_1 represents $x_1 + x_2$, z_2 represents x_4 , z_3 represents $x_7 + x_8 + 3x_{10}$, z_4 represents $x_{12} + x_{13}$.

Next, the ordering of the blocks $B_1, \dots, B_v$ is defined by $\Delta_1 = \emptyset$ and $\Delta_j = \{1, \dots, j-1\}$ for $j = 2, \dots, v$. Using this ordering, the inequality $I_4(396)$ obtained via our inductive scheme reads

$$1z_1 + 3z_2 + 3z_3 + 30z_4 \leq 103.$$

Finally, it suffices to transform this inequality into a valid inequality for the original polytope $\mathcal{P}_{SKP}$. This gives

$$1(x_1 + x_2) + 3(x_4) + 3(x_7 + x_8 + 3x_{10}) + 30(x_{12} + x_{13}) \leq 103.$$

The inequalities of the form (5) generalize other known inequalities. For the 0/1 sequential knapsack problem in the case that $k = 2$, the class of inequalities (5) reads

$$\sum_{i \in S_1 \cup T_1} x_i + \sum_{i \in N_2} |T_1| x_i \leq |S_1| + \left(\frac{F - |S_1|}{f_2} \right) |T_1|,$$

where $\emptyset \neq S_1 \subseteq N_1$, $T_1 \subseteq N_1 \setminus S_1$, $0 < |T_1| < f_2$, $|S_1| = F \pmod{f_2}$, and $|S_1| + f_2|N_2| > F$. It was shown in [11] and independently in [3] that this class together with the trivial inequalities $0 \leq x_i \leq 1$, $i \in N_1 \cup N_2$, the knapsack constraint, and the cover inequality $\sum_{i \in N_2} x_i \leq \lfloor \frac{F}{f_2} \rfloor$ describe the polytope $\text{conv}(\{x \in \{0, 1\}^{N_1 \cup N_2} : \sum_{i \in N_1} x_i + \sum_{i \in N_2} f_2 x_i \leq F, x \text{ integer}\})$.

As a special case one can derive Padberg's result on $(1, k)$ -configurations [8].

6. General 0/1 knapsack problems. Let us now briefly indicate how one can apply the polyhedral knowledge on the sequential knapsack polytope to more general cases. From an arbitrary 0/1 knapsack problem one can deduce a relaxation as a sequential knapsack polyhedron. An example of such a procedure is discussed next. Consider the constraints

$$\sum_{i \in N} f_i x_i \leq F, \quad x_i \in \{0, 1\} \text{ for } i \in N$$

with $N := \{1, \dots, s+1\}$, $s \geq 3$, and $f_1 \leq f_2 \leq \dots \leq f_s \leq f_{s+1}$. Let $S := \{1, \dots, s\}$ such that $f(S) \leq F$ and $f(S) + f_{s+1} > F$, and set $r := F - f(S)$.

We choose indices $1 = i_1 < i_2 < \dots < i_\tau < i_{\tau+1} = s+1$ such that $i_2 \geq 3$, $f_{i_j} \geq f_{i_{j-1}} + f_{i_{j-2}}$ for $j = 2, \dots, \tau$ and define a partition $S_1, S_2, \dots, S_{\tau+1}$ of the set N of items as

$$S_j := \{i_j, \dots, i_{j+1} - 1\} \text{ for } j = 1, \dots, \tau \text{ and } S_{\tau+1} = \{i_{\tau+1}\} = \{s+1\}.$$

Based on this partition, we define an inequality with the divisibility property that is valid for the given 0/1 knapsack problem. We set $b_1 := 1$ and, for $j = 2, \dots, \tau$,

$$t_j := \max \left\{ t \in \{1, \dots, i_j - i_{j-1}\} : \sum_{w=i_j-t}^{i_j-1} f_w \leq f_{i_j} \right\} \text{ and } b_j := b_{j-1} t_j.$$

Note that $2 \leq t_j \leq |S_{j-1}| = i_j - i_{j-1}$. Finally, we define $b_{\tau+1} := b_\tau t_{\tau+1}$ with

$$t_{\tau+1} := \max \left\{ t \in \{1, \dots, i_{\tau+1} - i_\tau\} : \sum_{w=i_{\tau+1}-t}^{i_{\tau+1}-1} f_w < f_{i_{\tau+1}} - r + f_1 \right\}.$$

If $f_s < f_{s+1} - r + f_1$ holds, then $t_{\tau+1}$ is well defined and we obtain the following proposition.

PROPOSITION 6.1. *If $f_s < f_{s+1} - r + f_1$, the inequality*

$$(6) \quad \sum_{j=1}^{\tau+1} b_j \sum_{i \in S_j} x_i \leq \sum_{j=1}^{\tau} b_j |S_j|$$

is valid for the 0/1 knapsack problem.

Example 6.2. Consider the knapsack constraint in 0/1 variables

$$x_1 + x_2 + 2x_3 + 3x_4 + 3x_5 + 3x_6 + 4x_7 + 7x_8 + 7x_9 + 7x_{10} + 8x_{11} + 25x_{12} \leq 49.$$

Set $S := \{1, \dots, 11\}$; then $f(S) = 46$ and $r = 3$. We choose $\tau = 3$, $i_1 = 1$, $i_2 = 4$, $i_3 = 8$, and $i_4 = 12$. This meets the requirements that the indices i_2, i_3 must satisfy because $f_4 \geq f_3 + f_2$ and $f_8 \geq f_7 + f_6$. In this example, inequality (6) is of the form

$$x_1 + x_2 + x_3 + 2x_4 + 2x_5 + 2x_6 + 2x_7 + 4x_8 + 4x_9 + 4x_{10} + 4x_{11} + 12x_{12} \leq 27,$$

and it is valid for the given knapsack polytope.

Proof of Proposition 6.1. Inequality (6) is valid if and only if every subset $T \subseteq S$ with $f(T) \geq f_{s+1} - r$ satisfies $b(T) := \sum_{j=1}^{\tau} b_j |S_j \cap T| \geq b_{\tau+1}$. This is equivalent to the statement that (6) is valid if and only if the problem

$$f^* := \max \left\{ \sum_{i=1}^s f_i y_i : \sum_{j=1}^{\tau} b_j \sum_{i \in S_j} y_i \leq b_{\tau+1} - 1, y \in \{0, 1\}^s \right\}$$

has an optimal value $f^* < f_{s+1} - r$.

Setting $Y_j := \sum_{i \in S_j} y_i$ for $j = 1, \dots, \tau$, we first show that there always exists an optimal solution to this problem with $Y_j < t_{j+1}$ for $j = 1, \dots, \tau$. First, observe that $Y_{\tau} \geq t_{\tau+1}$ is infeasible for this problem because $b_{\tau+1} = t_{\tau+1} b_{\tau}$, so $Y_{\tau} < t_{\tau+1}$. Suppose that $Y_{j-1} \geq t_j$ for some $j = 2, \dots, \tau$. By construction we have that $\sum_{w=i_j-t_j}^{i_j-1} f_w \leq f_{i_j}$ and $b_j = t_j b_{j-1}$. Hence, the solution obtained by decreasing Y_{j-1} by t_j and increasing Y_j by 1 is feasible and attains an objective function value that is not worse than the one of the initial solution. Applying these arguments for all $j = 2, \dots, \tau$ shows that every optimal solution with $Y_{j-1} \geq t_j$ for some $j \in \{2, \dots, \tau\}$ can be transformed into an optimal solution with $Y_{j-1} < t_j$ for all $j \in \{2, \dots, \tau\}$.

The objective value of such a solution satisfies

$$\sum_{i \in S_j} f_i y_i \leq f_{i_{j+1}} - f_{i_j} \quad \text{for } j = 1, \dots, \tau - 1$$

because $Y_j < t_{j+1} \leq |S_j|$ implies that there exists $z \in S_j$ with $y_z = 0$ and $f_z \geq f_{i_j}$ such that $f_{i_j} + \sum_{i \in S_j} f_i y_i \leq f_z + \sum_{i \in S_j} f_i y_i \leq f_{i_{j+1}}$.

Summing these inequalities for $j = 1, \dots, \tau - 1$, we obtain

$$\begin{aligned} \sum_{j=1}^{\tau} \sum_{i \in S_j} f_i y_i &= \sum_{j=1}^{\tau-1} \sum_{i \in S_j} f_i y_i + \sum_{i \in S_{\tau}} f_{\tau} y_{\tau} \\ &\leq -f_1 + \left(f_{i_{\tau}} + \sum_{i \in S_{\tau}} f_{\tau} y_{\tau} \right) \\ &\leq -f_1 + \left(\sum_{w=i_{\tau+1}-t_{\tau+1}}^{i_{\tau+1}-1} f_w \right) \\ &< f_{i_{\tau+1}} - r = f_{s+1} - r. \end{aligned}$$

Hence $f^* < f_{s+1} - r$, and the inequality is valid. $\square$

By construction, b_j is a multiple of b_{j-1} for all $j \geq 2$. It follows that (6) has the divisibility property, and we can apply all of our information for the sequential knapsack polytope induced by inequality (6). In the case that $\tau = 1$ and imposing

a “regularity condition” such as “every subset T in S with $b(T) = b(S) - b_{\tau+1}$ satisfies $f(T) + f_{s+1} \leq F$,” then the corresponding inequality defines a facet of the 0/1 knapsack polytope [8].

For $\tau \geq 2$, one can also derive sufficient conditions under which inequality (6) defines a facet of the corresponding polytope. Yet such conditions are quite technical and we refrain within this paper from explaining further details.

The idea of relaxing rows of a given 0/1-integer program by a sequential knapsack constraint in combination with the lifting of inequalities might help in solving integer programs.

7. Separation. A first question in using our inductively defined inequalities computationally is whether we have a combinatorial algorithm for solving the separation problem; i.e., given a fractional solution y , does there exist an inequality that is violated by y and, if so, then what is the inequality? We did not succeed in solving this separation problem.

Hartmann [3] gives a linear time algorithm for solving the separation problem associated with inequalities (5) (a subclass of our inductive inequalities) in the case that $k = 2$. Finding a polynomial combinatorial algorithm for solving the general separation problem is still open.

However, we can use our inductive scheme as a separation heuristic. For instance, defining every item $i \in N$ as a single block, setting $\Delta_i = \{t \in N : f_t \leq f_i, y_t > y_i\}$, $i \in N$, and generating an inequality according to this ordering might be a promising approach to end up with a violated inequality, if one exists. Other reasonable definitions of Δ_i might be to set $\Delta_i = \{t \in N : f_t \leq f_i, \frac{y_t}{f_t} > \frac{y_i}{f_i}\}$, $i \in N$. Whether those ideas work is certainly not clear, but similar “greedy types” of procedures work pretty well for the separation of cover- and $(1, k)$ -configuration inequalities.

Acknowledgments. We are very grateful to the referees of this paper, especially to the one who suggested Lemma 4.2 and its proof. Their suggestions helped to improve the readability of the paper.

REFERENCES

- [1] D. BOGART, M. SUDIT, AND T. OLMSTEAD, *The Sequential Knapsack Problem*, Working paper, Department of Industrial Engineering, SUNY, Buffalo, NY, 1991.
- [2] R. E. GOMORY, *Some polyhedra related to combinatorial problems*, Linear Algebra Appl., 2 (1969), pp. 551–558.
- [3] M. HARTMANN, *Cutting Planes and the Sequential Knapsack Problem*, Technical report TR-94/10, Univ. of North Carolina, Chapel Hill, NC, 1994.
- [4] M. HARTMANN AND T. OLMSTEAD, *Solving sequential knapsack problems*, Oper. Res. Lett., 13 (1993), pp. 225–232.
- [5] L. LOVÁSZ, *Graph theory and integer programming*, Ann. Discrete Math., 4 (1979), pp. 141–158.
- [6] O. MARCOTTE, *The cutting stock problem and integer rounding*, Math. Programming, 33 (1985), pp. 82–92.
- [7] M. W. PADBERG, *A Note on 0-1 Programming*, Oper. Res. 23 (1975), pp. 833–837.
- [8] M. W. PADBERG, *$(1, k)$ -Configurations and facets for packing problems*, Math. Programming, 18 (1980), pp. 94–99.
- [9] Y. POCHET AND L. WOLSEY, *Integer knapsack and flow covers with divisible coefficients: Polyhedra, optimization and separation*, Discrete Appl. Math., 59 (1995), pp. 57–74.
- [10] R. WEISMANTEL, *On the 0/1 knapsack polytope*, Math. Programming, 77 (1997), pp. 49–68.
- [11] R. WEISMANTEL, *Hilbert bases and the facets of special knapsack problems*, Math. Oper. Res., 21 (1996), pp. 886–904.
- [12] L. A. WOLSEY, *Faces of linear inequalities in 0-1 variables*, Math. Programming, 8 (1975), pp. 165–178.

AN ALGORITHM FOR THE INEQUALITY-CONSTRAINED DISCRETE MIN-MAX PROBLEM*

BERC RUSTEM[†] AND QUOC NGUYEN[†]

Abstract. In this paper we discuss an approach using an augmented Lagrangian formulation to directly solve the inequality constrained min-max problem. The algorithm involves a sequential quadratic programming subproblem, an adaptive penalty parameter selection rule, and a stepsize strategy, convergent to unit steps, that ensures progress toward optimality and feasibility of the inequality constraints. It is shown that the penalty parameter does not grow indefinitely. The convergence of the algorithm is established, and its numerical effectiveness is demonstrated with test examples.

Key words. rival scenarios, min-max, quasi-Newton algorithm, augmented Lagrangian, adaptive penalty, convergent stepsizes

AMS subject classifications. 90C30, 62C20, 90B50, 65K05

PII. S1056263493260386

1. Introduction. In financial decision making as well as in engineering design rival models of the underlying system, or rival future scenarios, often need to be taken into account. In Rustem (1987, 1992) we discuss a pooling of all rival possibilities as an extension of forecast pooling considered by Chow (1979), Fuhrer and Haltmaier (1986), Granger and Newbold (1977), Lawrence, Edmunson, and O'Connor (1986), and Makridakis and Winkler (1983). Let α denote the pooling weights, with

$$\alpha \in \mathbb{E}_+^m = \{\alpha \in \mathbb{R}^m \mid \alpha \geq 0; \langle 1, \alpha \rangle = 1\},$$

where $1 \in \mathbb{R}^m$ denotes the vector with every component unity. Consider the problem

$$(1.1) \quad \min_x \{\langle \alpha, f(x) \rangle \mid h(x) \leq 0\},$$

$x \in \mathbb{R}^n$. The functions $f: \mathbb{R}^n \rightarrow \mathbb{R}^m$; $h: \mathbb{R}^n \rightarrow \mathbb{R}^i$ are twice continuously differentiable and

$$f(x) = [f^1(x), \dots, f^j(x), \dots, f^m(x)]^T,$$

where f^j is the objective function representing the j th rival scenario or decision model. The vector α is fixed and defines the pooling weights. The restrictions $h(x) \leq 0$ are the inequality constraints on the overall problem.

Contrary to (1.1), where α is preselected, the min-max formulation (1.2) corresponds to the strategy that is invariant to whichever rival model actually turns out to represent the system

$$(1.2) \quad \min_x \max_{\alpha} \{\langle f(x), \alpha \rangle \mid h(x) \leq 0, \alpha \in \mathbb{E}_+^m\}.$$

The performance of the min-max decision, measured in terms of the corresponding objective value, is noninferior to (1.1) for the worst choice of α (i.e., the worst case) (Rustem (1987, Proposition 7.6)). Hence, (1.2) is a robust strategy.

*Received by the editors December 21, 1993; accepted for publication (in revised form) December 23, 1996.

<http://www.siam.org/journals/siopt/8-1/26038.html>

[†]Department of Computing, Imperial College of Science, Technology & Medicine, 180 Queen's Gate, London SW7 2BZ, UK (br@doc.ic.ac.uk).

In this paper, we discuss an augmented Lagrangian framework to directly solve the inequality constrained min-max problem. The algorithm integrates the method for equality constrained min-max in Rustem (1992) with the approach for inequality constraints in nonlinear programming in Rustem (1993). The algorithm in Rustem (1992) entails an equality constrained quadratic subproblem, a simple quadratic penalty function with a single penalty parameter, and a corresponding stepsize strategy. The present algorithm requires an inequality constrained quadratic subproblem: an adaptive penalty based on the function $(\max\{0, (\cdot)\})^2$ with a penalty adjustment factor for each constraint (Rustem (1993)). The penalty adjustment enables the replacement of $(\max\{0, (\cdot)\})^2$ by a smooth version at appropriate points, thereby ensuring twice continuous differentiability.

Thus, the min-max algorithm below requires a new quadratic programming subproblem based on the penalty, the objective functions, and the inequality constraints. The penalty parameter selection rule in Rustem (1993) is adapted to the discrete min-max problem. The stepsize strategy is based on a merit function measuring progress toward optimality and feasibility for the inequality constraints. This is achieved using the min-max objective in Rustem (1992) for the former and the penalty approach in Rustem (1993) for the latter. The descent property of the direction of search used by the algorithm, discussed in Lemma 3.4, is based on Lemmas 3.1–3.2, established in Rustem (1993), and the penalty parameter selection rule for the inequality constrained min-max algorithm. The min-max property in Lemma 3.5 is established by extending the arguments of Rustem (1992, Lemma 3.3) to the augmented Lagrangian of the inequality-constrained problem. The adaptive penalty parameter is shown not to grow indefinitely in Lemma 3.6 by extending the discussion in Rustem (1992, Lemma 3.4). To achieve this, the relation for inequality constraints in Lemma 3.1, established in Rustem (1993), is used with the optimality conditions of the inequality-constrained min-max quadratic subproblem. The monotonic decrease of the sequence of penalty function values is shown in Theorem 4.1 by using the property in Corollary 3.1, for the inequality constraint penalty function, and generalizing the equality-constrained min-max case in Rustem (1992, Theorem 4.1). Similarly, a basic convergence result is discussed in Lemma 4.1 by establishing the inequality-constrained case of a parallel result, for equality constraints, in Rustem (1992, Lemma 4.2). Finally, the global convergence of the method is established in Theorem 4.2. The convergence of the stepsize to unity and Q-superlinear convergence follow from the discussion related to the equality-constrained case (Rustem (1992)).

We discuss the Lagrangian and augmented Lagrangian for the discrete min-max problem in section 2. In section 3, we introduce the discrete min-max algorithm and its basic properties. In section 4, we establish its global convergence.

2. The discrete min-max approach. It can be shown that the min-max problem (1.2) is equivalent to

$$(2.1a) \quad \min_x \max_{j \in \{1, 2, \dots, m\}} \{f^j(x) \mid h(x) \leq 0\},$$

which, in turn, can be solved by the nonlinear programming problem

$$(2.1b) \quad \min_{x, v} \{v \mid f(x) \leq 1v; \quad h(x) \leq 0\},$$

where $v \in \mathbb{R}^1$ (Rustem (1992)).

Algorithms for solving the equality-constrained or -unconstrained min-max problems have been considered by a number of authors, including Charalambous and Conn

(1978), Coleman (1978), Conn (1979), Conn and Li (1992), Demyanov and Malozemov (1974), Demyanov and Pevnyi (1972), Dutta and Vidyasagar (1977), Hald and Madsen (1981), Han (1981), Murray and Overton (1980), Polak, Mayne, and Higgins (1991), Rustem (1992), and Womersley and Fletcher (1986).

The algorithm discussed below addresses the inequality-constrained problem by defining an appropriate penalty function and augmented Lagrangian. As in Rustem (1992), the penalty parameter is adaptively adjusted. However, in the inequality-constrained case, a further adjustment factor is introduced for each constraint. The resulting penalty function affects the inequality-constrained quadratic subproblem as well as the stepsize strategy. It is shown that the penalty parameter does not grow indefinitely. Convergence of the overall algorithm is established. Convergence to unit steplengths and Q-superlinear rate of convergence follow from results in Rustem (1992).

The augmented Lagrangian approach to the min-max problem is intended to ensure characterization of the solution of (1.2) as a saddle point of a convex Lagrangian function (Demyanov and Malozemov (1974), Arrow, Gould, and Howe (1973), Rockafeller (1973), and Arrow and Solow (1958)). This formulation enables the relaxation of the convexity requirements on the original minimization problem (see Arrow, Gould, and Howe (1973) and Rockafeller (1973)). For the equality-constrained min-max problem, a more detailed discussion is given by Rustem (1992).

We define a Lagrangian function for (1.2) as

$$L(x, \alpha, \lambda, \mu, \eta) = \langle f(x), \alpha \rangle + \langle h(x), \lambda \rangle - \langle \alpha, \mu \rangle + (\langle 1, \alpha \rangle - 1)\eta,$$

where $\lambda \in \mathbb{R}_+^i = \{\lambda \in \mathbb{R}^i \mid \lambda \geq 0\}$, $\mu \in \mathbb{R}_+^m = \{\mu \in \mathbb{R}^m \mid \mu \geq 0\}$, and $\eta \in \mathbb{R}^1$ are the multipliers associated with $h(x) \leq 0$, $\alpha \geq 0$, and $\langle 1, \alpha \rangle = 1$, respectively. An augmented Lagrangian for (1.2), is defined by

$$(2.2) \quad L^a(x, \alpha, \lambda, \mu, \eta, c, \pi) = L(x, \alpha, \lambda, \mu, \eta) + \frac{1}{2c} \|(ch(x) + \pi)_+\|_2^2,$$

where $0 \leq c \in \mathbb{R}_+^1$, $0 \leq \pi \in \mathbb{R}_+^i$, and $(\cdot)_+$ is such that its i th element is

$$(\cdot)_+^i = \max\{(\cdot)^i, 0\}.$$

It should be noted that (2.2) does not possess a continuous Hessian. In order to overcome the difficulties arising from this, we shall, at appropriate junctures, replace the Hessian of (2.2) with that of

$$(2.3) \quad \bar{L}^a(x, \alpha, \lambda, \mu, \eta, c, \pi) = L(x, \alpha, \lambda, \mu, \eta) + \frac{1}{2c} \|(ch(x) + \pi)\|_2^2.$$

In contrast to (2.2), the Hessian of $\bar{L}^a$ exists, provided f and h are suitably differentiable. The role of π in replacing the Hessian of (2.2) by that of (2.3) is crucial and resembles the *smearing* procedure adopted by Polak and Tits (1980). The difference is in the choice of π and c . It is shown below that this choice allows the treatment of the active constraints *locally* as equality constraints.

The penalty parameter c is adaptively determined as the algorithm progresses. Biggs (1974) and Polak and Tits (1981) give specific examples of this strategy in nonlinear programming. We adopt an adaptive strategy which departs from other works in the relationship of c with the descent property of the direction of search, as discussed in Lemmas 3.4 and 3.6. This is an extension of a nonlinear programming approach discussed in Rustem (1993). In particular, c is only adjusted to ensure that the direction of search is a descent direction for the penalty function (3.9).

3. The min–max algorithm. The algorithm generates a sequence of points $\{x_k\}$ given by $x_{k+1} = x_k + \tau_k d_k$, where d_k is the direction of search and τ_k is the stepsize along d_k . We discuss the determination of d_k in section 3.1, the adaptive determination of the penalty parameters π_k and c_k in section 3.2, the stepsize strategy in section 3.3, and the overall algorithm in section 3.4.

3.1. The quadratic programming subproblem for computing the search direction. The principal determinants of the search direction are linear approximations to the constraints and objective functions and an approximation to the Hessian of the augmented Lagrangian. For the linearization of the constraints, consider the matrix

$$\nabla h(x) = [\nabla h^1(x), \dots, \nabla h^i(x)].$$

Sometimes, $\nabla h(x)$ evaluated at x_k will be denoted by ∇h_k , and $h(x_k)$ will be denoted by h_k . Thus, a local linearization of $h(x)$ at x_k can be written as

$$h(x) = h_k + \nabla h_k^T [x - x_k].$$

Consider the function

$$\mathcal{F}(x, \alpha) = \langle \alpha, f(x) \rangle$$

and its linear approximation, with respect to x , at x_k ,

$$\mathcal{F}_k(x, \alpha) = \langle \alpha, f(x_k) + \nabla f(x_k)^T (x - x_k) \rangle,$$

where

$$\nabla f(x) = [\nabla f^1(x), \dots, \nabla f^m(x)].$$

We sometimes denote $f(x_k)$ and $\nabla f(x_k)$ by f_k and ∇f_k , respectively. Thus, for $d = x - x_k$, we have

$$(3.1) \quad \mathcal{F}_k(x_k + d, \alpha) = \langle \alpha, f_k + \nabla f_k^T d \rangle.$$

The quadratic objective used to compute the direction of progress is given by

$$\mathcal{F}_k(x_k + d, \alpha) + \langle d, \nabla h_k(c_k h_k + \pi_k)_+ \rangle + \frac{1}{2} \langle d, \hat{H}_k d \rangle.$$

The matrix $\hat{H}_k$ is a positive semidefinite¹ approximation to

$$(3.2) \quad H_k^a = \sum_{j=1}^m \alpha_k^j \nabla^2 f^j(x_k) + \sum_{i=1}^i \lambda_k^i \nabla^2 h^i(x_k) + c \nabla h_k \nabla h_k^T.$$

H_k^a is the Hessian of $\bar{L}^a$, excluding the second derivatives due to the penalty term in the augmented Lagrangian (Tapia (1986)). The values α_k and λ_k are given by the quadratic subproblem of the previous iteration. The direction of progress at each iteration is determined by the *quadratic subproblem*

$$(3.3a) \quad \min_d \max_{\alpha} \{ \mathcal{F}_k(x_k + d, \alpha) + \langle d, \nabla h_k(c_k h_k + \pi_k)_+ \rangle + \frac{1}{2} \langle d, \hat{H}_k d \rangle \mid \nabla h_k^T d + h_k \leq 0; \alpha \in \mathbb{E}_+^m \}.$$

¹The assumption of positive semidefiniteness will be tightened in the later stages of the discussion.

An equivalent problem to (3.3a) that is easier to solve and has fewer variables is the *quadratic programming subproblem* given by

$$(3.3b) \quad \min_{d,v} \{v + \langle d, \nabla h_k(c_k h_k + \pi_k)_+ \rangle + \frac{1}{2} \langle d, \hat{H}_k d \rangle \mid \nabla h_k^T d + h_k \leq 0, \nabla f_k^T d + f_k \leq 1v\}.$$

The equivalence can be verified by observing that the solution of either subproblem satisfies common optimality properties, as discussed in section 3.4. It can thus be shown that the values of α that solve (3.3a) are the multipliers associated with

$$\nabla f_k^T d + f_k \leq 1v$$

in (3.3b).

Assumption 3.1. The quadratic subproblem has a finite solution.

Assumption 3.1 implies that all the inequality constraints satisfied as equalities at the solution of (3.3) are linearly independent. This can be relaxed by increasing the complexity of the quadratic subproblem.

3.2. The penalty function. The penalty parameters to be determined are c and π . The basic role of π is to nullify the contribution of the penalty term if the corresponding constraint is satisfied. This ensures the existence of the Hessian of the augmented Lagrangian at specific stages. Let the i th element of the vectors h and π be respectively denoted by h^i and π^i . Let

$$(3.4a) \quad \pi_{k+\frac{1}{2}}^i = \begin{cases} -c_{k+1} h_k^i & \text{if } h_k^i \leq 0, \\ \pi_k^i & \text{if } h_k^i > 0 \end{cases}$$

and

$$(3.4b) \quad \pi_{k+1}^i = \begin{cases} -c_{k+1} h_{k+1}^i & \text{if } h_{k+1}^i \leq 0, \\ \pi_{k+\frac{1}{2}}^i & \text{if } h_{k+1}^i > 0, \end{cases}$$

which π_0 is set in **Step 0** in the algorithm in section 3.4. c_{k+1} is set in **Step 3**, with $c_0 \in (0, \infty]$ a given constant. The reason for distinguishing between π_{k+1} , $\pi_{k+1/2}$ is to ensure, in practice, that the first inequality in Corollary 3.1 is satisfied in general. It will be shown that the penalty parameter c_k need not be altered after reaching a finite value c_* . Thus, at the later stages of the algorithm we have $c_k = c_*$ and, for a constant c_* , it can be verified from (3.4) that $\pi_{k+1/2} = \pi_k$ since

$$\pi_k^i = \pi_{k+\frac{1}{2}}^i = \begin{cases} -c_* h_k^i & \text{if } h_k^i \leq 0, \\ \pi_k^i & \text{if } h_k^i > 0. \end{cases}$$

The determination of c_k is an important aspect of the algorithm and is linked with the merit function

$$(3.5) \quad \psi(x) = \max_{j \in \{1, 2, \dots, m\}} \{f^j(x)\}$$

and

$$\psi_k(x_k + d_k) = \max_{j \in \{1, 2, \dots, m\}} \{f^j(x_k) + \langle \nabla f^j(x_k), d_k \rangle\}.$$

Starting from $c_0 \in [0, \infty)$, if, for $\epsilon \in (0, \frac{1}{2}]$, the penalty parameter c_k satisfies the inequality

$$(3.6) \quad \begin{aligned} & \psi_k(x_k + d_k) - \psi(x_k) - c_k \|(h_k)_+\|_2^2 - \langle (h_k)_+, \pi_k \rangle \\ & + \left(\epsilon + \frac{1}{2} \right) \langle d_k, \hat{H}_k d_k \rangle \leq 0, \end{aligned}$$

then it remains unchanged $c_{k+1} = c_k$. Otherwise, it is increased to

$$(3.7) \quad c_{k+1} = \max \left\{ \frac{\psi_k(x_k + d_k) - \psi(x_k) + (\epsilon + \frac{1}{2}) \langle d_k, \hat{H}_k d_k \rangle - \langle (h_k)_+, \pi_k \rangle}{\|(h_k)_+\|_2^2}, c_k + \delta \right\}.$$

In section 3.4 it is shown that (3.6) is satisfied for finite c_k even when feasibility is achieved and $\|(h_k)_+\|_2^2 = 0$. Also, this choice of c_k is shown in Lemma 3.4 to ensure that d_k is a descent direction.

3.3. The stepsize strategy. Let the value of d , α , v solving (3.3) be denoted by d_k , α_{k+1} , v_{k+1} . The stepsize to be taken along d_k is determined using the equivalent min-max formulation (2.1) and the merit function (3.5). The stepsize strategy determines the smallest nonnegative integer t_k such that $\tau_k = \gamma^{t_k}$, $\gamma \in (0, 1)$, yields $x_{k+1} = x_k + \tau_k d_k$, which satisfies the inequality

$$(3.8a) \quad \begin{aligned} & \psi(x_{k+1}) + \frac{1}{2c_{k+1}} \|(c_{k+1}h_{k+1} + \pi_{k+1})_+\|_2^2 - \psi(x_k) - \frac{1}{2c_{k+1}} \|(c_{k+1}h_k + \pi_{k+\frac{1}{2}})_+\|_2^2 \\ & \leq \rho \tau_k \Phi(d_k, c_{k+1}, \pi_{k+\frac{1}{2}}) \end{aligned}$$

where $\rho \in (0, 1)$ is a given scalar and

$$(3.8b) \quad \begin{aligned} & \Phi(d_k, c_{k+1}, \pi_{k+\frac{1}{2}}) = \psi_k(x_k + d_k) - \psi(x_k) + \langle d_k, \nabla h_k(c_{k+1}h_k + \pi_{k+\frac{1}{2}})_+ \rangle \\ & + \frac{1}{2} \langle d_k, \hat{H}_k d_k \rangle. \end{aligned}$$

The stepsize τ_k determined by (3.8) ensures that x_{k+1} simultaneously improves the objective function and the feasibility with respect to the constraints. The penalty term used to measure feasibility is a quadratic consistent with the augmented Lagrangian (2.3). It is shown in Theorem 4.1 that (3.8) can always be fulfilled by the algorithm.

3.4. The algorithm.

- Step 0:** Given x_0 , $c_0 \in [0, \infty)$, and small positive numbers δ , ρ , ϵ , γ such that $\delta \in (0, \infty)$, $\rho \in (0, 1)$, $\epsilon \in (0, \frac{1}{2}]$, $\gamma \in (0, 1)$, $\hat{H}_0$, set $\pi_0^i = \max\{-c_0 h^i(x_0), 0\}$, $i = 1, \dots, i$, $k = 0$.
- Step 1:** Compute ∇f_k and ∇h_k . Solve the quadratic subproblem (3.3b) (adopting subproblem (3.3a) also yields an equivalent algorithm) to obtain d_k , α_{k+1} , v_{k+1} , λ_{k+1} .
- Step 2:** Test for optimality. If optimality is achieved, stop. Else go to **Step 3**.
- Step 3:** If (3.6) is satisfied, then $c_{k+1} = c_k$. Else compute c_{k+1} given by (3.7).
- Step 4:** Find the smallest nonnegative integer t_k such that $\tau_k = \gamma^{t_k}$ with $x_{k+1} = x_k + \tau_k d_k$ such that the inequality (3.8) is satisfied.

Step 5: Update $\hat{H}_k$ to compute $\hat{H}_{k+1}$, set $k = k + 1$, and go to **Step 1**.

In **Step 3**, the penalty parameter c_{k+1} is adjusted to ensure that progress towards feasibility is maintained. In particular, c_{k+1} is chosen to make d_k a descent direction for the penalty function

$$(3.9) \quad \psi(x_k) + \frac{1}{2c_{k+1}} \left\| (c_{k+1}h_k + \pi_{k+\frac{1}{2}})_+ \right\|_2^2.$$

We state the basic properties of the penalty term. The reader is referred to Rustem (1993) for the proofs.

LEMMA 3.1. *Let $\kappa \geq 0$ be a constant, $c_{k+1} \geq 0$, and*

$$\pi_{k+1}^i(x) = \begin{cases} -c_{k+1}h^i(x) & \text{if } h^i(x) \leq 0, \\ \kappa & \text{if } h^i(x) > 0. \end{cases}$$

Thus, we have

$$(c_{k+1}h^i(x) + \pi_{k+1}^i(x))_+ = (c_{k+1}h^i(x) + \pi_{k+1}^i(x)) \geq 0.$$

LEMMA 3.2. *We have*

$$\langle (h_k)_+, \pi_{k+\frac{1}{2}} \rangle = \langle (h_k)_+, \pi_k \rangle.$$

The following corollary ensures that the descent discussion in Lemma 3.4 and Theorem 4.1 can always be related to the stepsize strategy in **Step 4** when the penalty parameter is not necessarily a constant. Starting with a penalty term which is not differentiable, we use the algorithm definition of π to ensure the twice differentiability of $\|(c_{k+1}h_k + \pi_{k+1/2})\|_2^2$. This leads to the convergence results of the next section.

COROLLARY 3.1. *We have*

$$\begin{aligned} & \|(c_{k+1}h_{k+1} + \pi_{k+1})_+\|_2^2 - \|(c_{k+1}h_k + \pi_{k+\frac{1}{2}})_+\|_2^2 \\ & \leq \|(c_{k+1}h_{k+1} + \pi_{k+\frac{1}{2}})_+\|_2^2 - \|(c_{k+1}h_k + \pi_{k+\frac{1}{2}})_+\|_2^2 \\ & \leq \|(c_{k+1}h_{k+1} + \pi_{k+\frac{1}{2}})\|_2^2 - \|(c_{k+1}h_k + \pi_{k+\frac{1}{2}})_+\|_2^2 \\ & = \|(c_{k+1}h_{k+1} + \pi_{k+\frac{1}{2}})\|_2^2 - \|(c_{k+1}h_k + \pi_{k+\frac{1}{2}})\|_2^2. \end{aligned}$$

We summarize the optimality conditions of the quadratic subproblem (3.3). Subsequently, we establish the descent property of d_k and that c_k determined by (3.7) is not increased indefinitely.

LEMMA 3.3. *The Lagrangian function for the quadratic subproblem (3.3a) is given by*

$$\begin{aligned} L_k(x, \alpha, \lambda, \mu, \eta, c_k, \pi_k) &= \mathcal{F}_k(x_k + d, \alpha) + \langle d, \nabla h_k(c_k h_k + \pi_k)_+ \rangle \\ (3.10a) \quad &+ \frac{1}{2} \langle d, \hat{H}_k d \rangle + \langle \lambda, \nabla h_k^T d + h_k \rangle \\ &- \langle \alpha, \mu \rangle + (\langle 1, \alpha \rangle - 1)\eta. \end{aligned}$$

The first-order necessary conditions of optimality for problem (3.3a) are

$$(3.10b) \quad \nabla f_k \alpha_{k+1} + \nabla h_k[(c_k h_k + \pi_k)_+ + \lambda_{k+1}] + \hat{H}_k d_k = 0,$$

$$(3.10c) \quad f_k + \nabla f_k^T d_k - I\mu_{k+1} + 1\eta_{k+1} = 0,$$

$$(3.10d) \quad \alpha_{k+1} \in \mathbb{E}_+^m, \quad \mu_{k+1} \geq 0, \quad \alpha_{k+1}^j \mu_{k+1}^j = 0 \quad \forall j \in \{1, \dots, m\},$$

$$(3.10e) \quad \nabla h_k^T d_k + h_k \leq 0, \quad \langle \lambda_{k+1}, \nabla h_k^T d_k + h_k \rangle = 0, \quad \lambda_{k+1} \geq 0.$$

For the quadratic programming subproblem (3.3b) the necessary conditions are given by (3.10b), (3.10e), and

$$(3.10f) \quad f_k + \nabla f_k^T d_k \leq 1v_{k+1},$$

$$\alpha_{k+1}^j (f_k^j + \langle \nabla f_k^j, d_k \rangle - v_{k+1}) = 0, \quad j = 1, \dots, m, \alpha_{k+1} \in \mathbb{E}_+^m.$$

LEMMA 3.4 (the descent property of d_k). *Let $(h_k)_+ \neq 0$ (the case when $(h_k)_+ = 0$ is discussed in Lemma 3.6), and let $\hat{H}_k$ be positive semidefinite. The direction d_k satisfying the linearized inequality constraints (3.10e) of subproblem (3.3) and c_{k+1} chosen as in **Step 3** of the above algorithm ensure the inequality*

$$(3.11) \quad \Phi(d_k, c_{k+1}, \pi_{k+\frac{1}{2}}) \leq -\epsilon \langle d_k, \hat{H}_k d_k \rangle.$$

Proof. We note that Lemma 3.1 equally applies to π_k and $\pi_{k+1/2}$. Using (3.10e), Lemma 3.1, and Lemma 3.2, we can write

$$\begin{aligned} \langle d_k, \nabla h_k(c_{k+1}h_k + \pi_{k+\frac{1}{2}})_+ \rangle &\leq -\langle h_k, (c_{k+1}h_k + \pi_{k+\frac{1}{2}})_+ \rangle \\ &= -c_{k+1} \|(h_k)_+\|_2^2 - \langle (h_k)_+, \pi_k \rangle. \end{aligned}$$

c_{k+1} chosen in **Step 3** ensures that (3.8b) is bounded by

$$(3.12) \quad \begin{aligned} \Phi(d_k, c_{k+1}, \pi_{k+\frac{1}{2}}) &\leq \psi_k(x_k + d_k) - \psi(x_k) \\ &\quad - c_{k+1} \|(h_k)_+\|_2^2 - \langle (h_k)_+, \pi_k \rangle + \frac{1}{2} \langle d_k, \hat{H}_k d_k \rangle \\ &\leq -\epsilon \langle d_k, \hat{H}_k d_k \rangle. \end{aligned}$$

Thereby (3.11) is established. $\square$

LEMMA 3.5. *Let d_k and α_{k+1} be the solutions of the quadratic subproblem (3.3); then*

$$\psi_k(x_k + d_k) = \max_{\alpha \in \mathbb{E}_+^m} \mathcal{F}_k(x_k + d_k, \alpha) = \mathcal{F}_k(x_k + d_k, \alpha_{k+1}).$$

Proof. The proof follows mostly from Rustem (1992, Lemma 3.3). In the inequality-constrained case, we need to consider (3.10a) or its equivalent for subproblem (3.3b). For $(d_k, \lambda_{k+1}, \mu_{k+1}, \eta_{k+1}, v_{k+1})$ solving (3.3) we have

$$(3.13) \quad \begin{aligned} \alpha_{k+1} &= \arg \max_{\alpha \in \mathbb{E}_+^m} L_k(x_k + d_k, \alpha, \lambda_{k+1}, \mu_{k+1}, \eta_{k+1}, c_k, \pi_k) \\ &= \arg \max_{\alpha \in \mathbb{E}_+^m} \mathcal{F}_k(x_k + d_k, \alpha). \end{aligned}$$

Thus, α_{k+1} maximizes both L_k and $\mathcal{F}_k$ at $(d_k, \lambda_{k+1}, \mu_{k+1}, \eta_{k+1})$. $\square$

It is shown in Lemma 3.6 that (3.12) is nonpositive and the descent property of Lemma 3.4 also holds when $(h_k)_+ = 0$. The proof follows similar arguments as in Rustem (1992, Lemma 3.4) except that subproblems (3.3a) and (3.3b), for the inequality-constrained case, are used with associated penalty functions and parameters.

LEMMA 3.6 (the finiteness of the penalty parameter). *Let*

- (i) $f, h \in \mathbb{C}^1(\mathbb{R}^n)$;
- (ii) $\hat{H}_k$ be positive semidefinite; and
- (iii) $(d_k, v_{k+1}, \alpha_{k+1}, \lambda_{k+1}) \in \mathbb{R}^n \times \mathbb{R}^1 \times \mathbb{R}^m \times \mathbb{R}^i$, computed by (3.3b),² be bounded.³

Then

(a) *If the nonlinear constraints of (1.2) are satisfied (i.e., $h(x_k) \leq 0$), then the descent property (3.11) of Lemma 3.4 is satisfied for any choice of $c_k \in [0, \infty)$.*

(b) *Let $c_k \in [0, \infty)$ and $h(x_k) \leq 0$ for some k . Then **Step 3** of the algorithm chooses $c_{k+1} = c_k$. For this choice of c_{k+1} , the direction d_k ensures the descent property (3.11).⁴*

(c) $\forall k, \exists c_{k+1} \in [0, \infty)$ satisfying **Step 3** of the algorithm.

(d) *Let the sequence $\{x_k\}$ generated by the algorithm be bounded. Then, c_k is increased finitely often and there exists an integer $k_* \geq 0$ and $\exists c_* \in [0, \infty)$ such that $c_k = c_* \forall k \geq k_*$.*

Proof. To show (a), consider (3.10b), (3.10e), and (3.4) for $h_k \leq 0$. By Lemma 3.1,

$$\begin{aligned} 0 &= \langle d_k, [\nabla f_k \alpha_{k+1} + \nabla h_k[(c_k h_k + \pi_k)_+ + \lambda_{k+1}] + \hat{H}_k d_k] \rangle \\ &= \langle d_k, \nabla f_k \alpha_{k+1} \rangle - \langle h_k, \lambda_{k+1} \rangle + \langle d_k, \hat{H}_k d_k \rangle \end{aligned}$$

and, hence, we have

$$(3.14) \quad \langle d_k, \nabla f_k \alpha_{k+1} \rangle = -\langle d_k, \hat{H}_k d_k \rangle.$$

Consider (3.12) for $h_k \leq 0$

(3.15)

$$\begin{aligned} &\Phi(d_k, c_{k+1}, \pi_{k+\frac{1}{2}}) + \epsilon \langle d_k, \hat{H}_k d_k \rangle \\ &= \psi_k(x_k + d_k) - \psi(x_k) + \langle d, \nabla h_k(c_{k+1} h_k + \pi_{k+\frac{1}{2}})_+ \rangle + \left(\frac{1}{2} + \epsilon\right) \langle d_k, \hat{H}_k d_k \rangle \\ &= \max_{\alpha \in \mathbb{E}_+^m} \langle \alpha, f_k + \nabla f_k^T d_k \rangle - \max_{\alpha \in \mathbb{E}_+^m} \langle \alpha, f_k \rangle + \left(\frac{1}{2} + \epsilon\right) \langle d_k, \hat{H}_k d_k \rangle \\ &= \langle \alpha_{k+1}, f_k + \nabla f_k^T d_k \rangle - \max_{\alpha \in \mathbb{E}_+^m} \langle \alpha, f_k \rangle + \left(\epsilon + \frac{1}{2}\right) \langle d_k, \hat{H}_k d_k \rangle \\ &\leq \left(\epsilon - \frac{1}{2}\right) \langle d_k, \hat{H}_k d_k \rangle \\ &\leq 0. \end{aligned}$$

²For subproblem (3.3a), $(d_k, \alpha_{k+1}, \lambda_{k+1}, \mu_{k+1}, \eta_{k+1}) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^i \times \mathbb{R}^m \times \mathbb{R}^1$ is required to be bounded.

³Along with assumptions (i) and (ii), boundedness is ensured when the active constraint normals at the solution of the quadratic subproblem are linearly independent. Another condition implied by (iii) is that the feasible set of the quadratic subproblem is nonempty.

⁴ c_k is the value used in the quadratic subproblems when computing $d_k, \alpha_{k+1}, \lambda_{k+1}, v_{k+1}$.

Equality (3.15) follows from Rustem (1992, Lemma 3.4). The subsequent equality follows from Lemma 3.5. Thus, the descent condition (3.11) is satisfied for $h_k \geq 0$.

To show (b), consider the condition (3.6) which is satisfied for all $c_k \in [0, \infty)$ since descent is ensured for $h_k \leq 0$. Thus, the algorithm chooses $c_{k+1} = c_k$. For this choice, it can simply be verified that Lemma 3.4 and (a) hold.

The proofs of (c) and (d) follow Rustem (1992, Lemma 3.4). $\square$

The algorithm increases the penalty parameter until c_* is attained. An increase of the penalty parameter is not required after this since the condition in **Step 3** is satisfied with $c_k = c_*$.

4. Convergence of the algorithm. We consider the global convergence of the method. Inequality constraints require a substantially different approach to the equality-constrained case discussed in Rustem (1992). Convergence of the stepsize sequence $\{\tau_k\}$ to unity and the superlinear rate of convergence are consequences of results in Rustem (1992, 1993).

We first establish that the stepsize strategy (3.8) can always be satisfied. If the satisfaction of (3.8) was assumed, then the final inequality in the proof, given by (4.8), could be replaced by (3.8). Lemmas 3.4 and 3.6(i) would provide the monotonicity result of Theorem 4.1.

Assumption 4.1. $f, h \in \mathbb{C}^2(\mathbb{R}^n)$.

THEOREM 4.1. *Let*

- (i) *Assumption (4.1) be satisfied;*
- (ii) *the approximate Hessian be such that, for each k ;*

$$(4.1) \quad m\|y\|_2^2 \leq y^T \hat{H}_k y \leq M\|y\|_2^2; \quad M \geq m > 0; \quad 0 \neq y \in \mathbb{R}^n;$$

- (iii) *Assumption (iii) of Lemma 3.6 be satisfied; and,*

- (iv) *there exist k_* and $c_* \geq 0$ such that (3.6) is satisfied with $c_k = c_*$ for all $k \geq k_*$.*

Then there exists $\tau_k \in (0, 1]$ that satisfies (3.8) and $\{x_k, \alpha_k, \Pi_k\}$, $k \geq k_*$, generates a monotonically decreasing sequence

$$\left\{ \psi(x_k) + \frac{1}{2c_*} \|(c_* h_k + \pi_k)_+\|_2^2 \right\}.$$

Remark. By Lemmas 3.4 and 3.6, c_k chosen by **Step 3** is finite. Also, $\exists c_* \in [0, \infty)$, for which (3.6) is satisfied with $c_k = c_*$ $\forall k \geq k_*$. The integer k_* is the iteration at which c_* is attained. The result can be extended to show that there exists $\tau_k \in (0, 1]$ satisfying the stepsize strategy *before* the algorithm achieves $c_k = c_*$. This requires the expansion of $\frac{1}{2c_{k+1}} \|(c_{k+1} h(x) + \pi_{k+1/2})_+\|_2^2$ instead of $\frac{1}{2c_*} \|(c_* h(x) + \pi_k)_+\|_2^2$. The assumption of $c_k = c_*$ is required to establish the overall monotonicity.

Proof. For $c_k = c_*$, (3.4) yields $\pi_{k+1/2} = \pi_k$. Using this in Corollary 3.1 and considering the second order expansion of $\frac{1}{2c_*} \|(c_* h(x) + \pi_k)_+\|_2^2$, we have

$$(4.2) \quad \begin{aligned} & \frac{1}{2c_*} \|(c_* h_{k+1} + \pi_{k+1})_+\|_2^2 - \frac{1}{2c_*} \|(c_* h_k + \pi_k)_+\|_2^2 \\ & \leq \frac{1}{2c_*} \|(c_* h_{k+1} + \pi_k)_+\|_2^2 - \frac{1}{2c_*} \|(c_* h_k + \pi_k)_+\|_2^2 \\ & \leq \tau_k \left\langle d_k, \nabla h_k(c_* h_k + \pi_k) + \frac{\tau_k}{2} \hat{H}_k d_k \right\rangle + \tau_k^2 \nu_k \|d_k\|_2^2, \end{aligned}$$

where, for $x(t) = x_k + \tau_k d_k$,

$$\nu_k = \int_0^1 (1-t) \left\| c_* \nabla h(x(t)) \nabla h(x(t))^T + \sum_{i=1}^i \nabla^2 h^i(x(t)) [c_* h^i(x(t)) + \pi_k^i] - \hat{H}_k \right\| dt.$$

By definition of $\psi(x)$ we have

$$\begin{aligned} \psi(x_{k+1}) &= \max_{\alpha \in \mathbb{E}_+^m} \{ \langle \alpha, f(x_k + \tau_k d_k) \rangle \} \\ &\leq \max_{\alpha \in \mathbb{E}_+^m} \{ \langle \alpha, f(x_k) + \tau_k \nabla f_k^T d_k \rangle \} \\ (4.3) \quad &+ \tau_k^2 \left\langle d_k, \left[\int_0^1 (1-t) \max_{\alpha \in \mathbb{E}_+^m} \left\{ \sum_{j=1}^m \alpha^j \nabla^2 f^j(x(t)) \right\} dt \right] d_k \right\rangle. \end{aligned}$$

Using (3.1) we can write

$$(4.4) \quad \max_{\alpha \in \mathbb{E}_+^m} \{ \langle \alpha, f(x_k) + \tau_k \nabla f_k^T d_k \rangle \} = \max_{\alpha \in \mathbb{E}_+^m} \mathcal{F}_k(x_k + \tau_k d_k, \alpha).$$

Since $\max_{\alpha \in \mathbb{E}_+^m} \mathcal{F}_k(x_k + d, \alpha)$ is convex with respect to d , we have for $\tau_k \in [0, 1]$

$$\begin{aligned} &\max_{\alpha \in \mathbb{E}_+^m} \mathcal{F}_k(x_k + \tau_k d_k, \alpha) \\ (4.5) \quad &\leq \max_{\alpha \in \mathbb{E}_+^m} \mathcal{F}_k(x_k, \alpha) + \tau_k \left\{ \max_{\alpha \in \mathbb{E}_+^m} \mathcal{F}_k(x_k + d_k, \alpha) - \max_{\alpha \in \mathbb{E}_+^m} \mathcal{F}_k(x_k, \alpha) \right\}. \end{aligned}$$

Furthermore, we have

$$(4.6a) \quad \psi(x_k) = \max_{\alpha \in \mathbb{E}_+^m} \mathcal{F}_k(x_k, \alpha),$$

$$(4.6b) \quad \psi_k(x_k + d) = \max_{\alpha \in \mathbb{E}_+^m} \mathcal{F}_k(x_k + d, \alpha).$$

Using (4.6), (4.5), (4.4), and Lemmas 3.4 and 3.6(i) and combining (4.2) and (4.3) yield

$$\begin{aligned} &\psi(x_{k+1}) + \frac{1}{2c_*} \|(c_* h_{k+1} + \pi_{k+1})_+\|_2^2 \\ (4.7) \quad &\leq \psi(x_k) + \frac{1}{2c_*} \|(c_* h_k + \pi_k)_+\|_2^2 + \tau_k \Phi(d_k, c_*) \left\{ 1 - \frac{\tau_k}{\epsilon m} \xi_k \right\}, \\ &\xi_k = \int_0^1 (1-t) \left\| \max_{\alpha \in \mathbb{E}_+^m} \left\{ \sum_{j=1}^m \alpha^j \nabla^2 f^j(x(t)) \right\} + c_* \nabla h(x(t)) \nabla h(x(t))^T \right. \\ &\quad \left. + \sum_{i=1}^i \nabla^2 h^i(x(t)) [c_* h^i(x(t)) + \pi_k^i] - \hat{H}_k \right\| dt. \end{aligned}$$

The scalar $\rho \in (0, 1)$ in the stepsize strategy (3.8) determines τ_k such that

$$\rho \leq 1 - \frac{\tau_k}{\epsilon m} \xi_k \leq 1.$$

By Lemmas 3.4 and 3.6(i), the descent property $\Phi(d_k, c_*, \pi_k) \leq 0$ holds. Thus, there exists $\tau_k \in (0, 1]$ to ensure (4.7); and thus (3.8). Suppose τ^0 is the largest $\tau \in (0, 1]$ satisfying (4.7) and (3.8). It follows that all $\tau \leq \tau^0$ also satisfy these conditions and the strategy in **Step 4** selects $\tau_k \in [\gamma\tau^0, \tau^0]$. The required monotonic decrease follows from $\Phi(d_k, c_*, \pi_k) \leq 0$. $\square$

LEMMA 4.1. *Let*

- (i) *the assumption of Theorem 4.1 be satisfied; and*
- (ii) *for some k_0 and $k \geq k_0$, the set*

$$\begin{aligned} \mathbb{F} = & \left\{ x \in \mathbb{R}^n \left| \psi(x) + \frac{1}{2c_*} \|(c_*h(x) + \pi_k(x))_+\|_2^2 \right. \right. \\ & \left. \left. \leq \psi(x_{k_0}) + \frac{1}{2c_*} \|(c_*h(x_{k_0}) + \pi_{k_0}(x_{k_0}))_+\|_2^2 \right\} \right\} \end{aligned}$$

be bounded.

We then have

$$(4.8) \quad \lim_{k \rightarrow \infty} \Phi(d_k, c_*, \pi_k) = 0.$$

Proof. For $c_k = c_*$, we have, for a constant $\kappa \geq 0$,

$$\pi_k^i(x) = \begin{cases} -c_*h^i(x) & \text{if } h^i(x) \leq 0, \\ \kappa & \text{if } h^i(x) > 0. \end{cases}$$

Given $\rho \in (0, 1)$, by (4.7), the choice $\tau^0 = \min\{1, \frac{(1-\rho)\epsilon m}{\xi_k}\}$ always satisfies (3.8). Clearly, $\tau_k = \gamma\tau^0$ chosen in **Step 4** is in the range $\tau_k \in [\gamma\tau^0, \tau^0]$ and thereby also satisfies (3.8). As f and h are twice continuously differentiable and $\mathbb{F}$ is compact, there is a scalar $\bar{M} < \infty$ such that $\xi_k \leq \bar{M}$. Thus, as $m > 0$, we have $\tau_k \geq \bar{\tau} > 0 \forall k$ for (3.8). The boundedness of

$$\psi(x) + \frac{1}{2c_*} \|(c_*h(x) + \pi_k(x))_+\|_2^2$$

on $\mathbb{F}$ and $\Phi(d_k, c_*, \pi_k) \leq 0$ imply

$$\begin{aligned} 0 & \leq \rho \sum_k \tau_k |\Phi(d_k, c_*, \pi_k)| \\ & \leq \sum_k \left[\psi(x_k) + \frac{1}{2c_*} \|(c_*h_k + \pi_k)_+\|_2^2 \right] \\ & \quad - \left[\psi(x_{k+1}) + \frac{1}{2c_*} \|(c_*h_{k+1} + \pi_{k+1})_+\|_2^2 \right] \\ & < \infty, \end{aligned}$$

which yields (4.8). $\square$

LEMMA 4.2. *Let the assumptions of Theorem 4.1 be satisfied. Inequality (3.11) and Lemmas 3.6(i) and 4.1 imply $\lim_{k \rightarrow \infty} \|d_k\| = 0$.*

Proof. Using (4.8) with (3.11) yields the required result. $\square$

THEOREM 4.2. *Let*

- (i) *the assumptions of Theorem 4.1 be satisfied;*
- (ii) *the first order necessary conditions of problem (2.1) be satisfied by $(x_*, \alpha_*, \lambda_*, v_*)$; and*
- (iii) *for large k , strict complementarity hold at the solution of (3.3b).*

Then

- (a) *the algorithm either terminates at $x_*, \alpha_*, \lambda_*, v_*$ or generates an infinite sequence $\{x_k\}$ with a subsequence $\{x_k\}$, $k_* \leq k \in K \subset \{0, 1, \dots\}$, such that $\{\|d_k\|\} \rightarrow 0$ and every accumulation point of the infinite sequence is x_* ; and*
- (b) *for large k , (α_k, λ_k) predicts the active inequality constraints at x_* .*

Proof. By Lemmas 3.6, 4.1, and 4.2, there exists a subsequence $\{x_k\}$ such that $\{\|d_k\|\} \rightarrow 0$. Let there exist $x_* \in \mathbb{R}^n$, $\alpha_* \in \mathbb{E}_+^m$, $\lambda_* \in \mathbb{R}_+^i$, $v_* \in \mathbb{R}^1$ such that $\{x_k\} \rightarrow x_*$, $\{\alpha_k\} \rightarrow \alpha_*$, $\{\lambda_k\} \rightarrow \lambda_*$, $\{v_k\} \rightarrow v_*$, $k_* \leq k \in K$. This is ensured since, by Theorem 4.1 the algorithm decreases

$$\psi(x_k) + \frac{1}{2c_*} \|(c_* h_k + \pi_k)_+\|_2^2$$

sufficiently at each iteration. Hence we have $x_k \in \mathbb{F}$, with $\mathbb{F}$ compact. The rest of the proof follows from Rustem (1992, Theorem 4.2, and 1993, Theorem 3.2). $\square$

The convergence of the stepsizes to unity and the Q-superlinear convergence rate of the algorithm can be studied by invoking Theorem 4.2(ii). By considering the inequality constraints active at the solution, the results for the equality-constrained problem, in Rustem (1992), are applicable. The convergence of the stepsize τ_k to unity is related to the accuracy of the Hessian approximation used in the quadratic subproblem. It is shown that the one-sided-projected-Hessian condition ensures the Q-superlinear convergence of $\{x_k\}$ and $\{x_k, \alpha_k, v_k\}$ and the two step Q-superlinear convergence of $\{x_k, \alpha_k, \lambda_k, v_k\}$. An extended one-sided-projected-Hessian *necessary* and *sufficient* condition is also established for the Q-superlinear convergence of $\{x_k\}$ and $\{x_k, v_k\}$.

5. Numerical experiments. We present a number of numerical examples to demonstrate the performance of the min-max algorithm. A comparison is also provided with a general nonlinear programming algorithm by converting the min-max problem (2.1a) into (2.1b). The nonlinear program used is the subroutine E04VDF of NAG, implementing the sequential quadratic programming algorithm due to Gill, Murray, and Wright (1981). The quadratic program used in the min-max algorithm is the subroutine E04NAF of NAG. The results are given in Table 1.

All the computations were performed in double precision on a SPARC 2 machine. The parameters for the algorithm were set to $\delta = 0.1$, $\rho = 0.1$, and $\epsilon = 0.0$. We present two sets of results for the constant regulating the stepsize strategy, $\gamma = 0.1$ and $\gamma = 0.5$. The stopping criterion for convergence was taken to be $\|d_k\|_2 \leq 1.0 \cdot 10^{-6}$.

All approximate Hessian computations of the augmented Lagrangian were done using Powell's (1978) modification of the BFGS formula. In initializing the Hessian, the problem was solved using the identity matrix and also using a numerical estimate of the Hessian, assuming zero initial constraint Lagrange multipliers and $\alpha_0^j = \frac{1}{m}$ (hence all α_0^j are equal and $\sum_{j=1}^m \alpha_0^j = 1.0$). The result reported corresponds to the

TABLE 1

	NAG		Min-Max algorithm ($\gamma = 0.1$)						Min-Max algorithm ($\gamma = 0.5$)					
	Itns	Evals	Itns	Evals	c_0	c_*	k_{c_*}	$k_{\tau=1}$	Itns	Evals	c_0	c_*	k_{c_*}	$k_{\tau=1}$
P1(A)	6	7	5	5	0.0	0.0	1	1	5	5	0.0	0.0	1	1
P1(B)	9	12	6	6	0.0	0.0	1	1	6	6	0.0	0.0	1	1
P1(C)	8	10	6	6	0.0	0.0	1	1	6	6	0.0	0.0	1	1
P1(D)	20	34	16	16	0.0	0.0	1	1	16	16	0.0	0.0	1	1
P2(A)	22	28	17	17	10.0	10.0	1	1	17	17	10.0	10.0	1	1
P2(B)	14	18	7	7	10^{-2}	10^{-2}	1	1	7	7	0.01	0.01	1	1
P3(A)	F ¹		13	13	0.0	0.0	1	1	13	13	0.0	0.0	1	1
P3(B)	18	20 ¹	9	9 ²	0.0	0.0	1	1	9	9 ²	0.0	0.0	1	1
P4(A) ³	11	15	8	10	1.0	23.583	2	3	7	8	1.0	22.395	1	2
P4(B) ⁴	11	21	8	9	1.0	33.424	1	2	7	10	1.0	33.424	1	2
P4(C) ⁴	F ²		9	10	1.0	14.911	1	2	8	9	1.0	14.911	1	2
P4(D) ³	10	11	10	11	1.0	4.583	2	2	10	11	1.0	8.184	2	2
P5(A)	17	32	10	12	0.0	0.0	1	9	10	11	0.0	0.0	1	3
P5(B)	10	15	11	14	0.0	0.0	1	11	10	13	0.0	0.0	1	10
P5(C)	21	34	10	10	0.0	0.0	1	1	10	10	0.0	0.0	1	1
P5(D)	17	24	10	10	0.0	0.0	1	1	10	10	0.0	0.0	1	1
P5(E)	18	28	10	11	0.0	0.0	1	9	10	10	0.0	0.0	1	9
P6(A)	5	6	4	4	0.0	0.0	1	1	4	4	0.0	0.0	1	1
P6(B)	10	12	7	7	0.0	0.0	1	1	7	7	0.0	0.0	1	1
P7(A)	23	40	17	27	0.0	0.0	1	10	15	33	0.0	0.0	1	8
P7(B)	F ³		20	30	0.0	0.0	1	14	18	42	0.0	0.0	1	10
P8(A)	9	11	29	62	10^{-5}	10^{-5}	1	27	14	46	16^{-5}	10^{-5}	1	12
P9(A)	11	12	9	9	0.0	0.0	1	1	8	8	0.0	0.0	1	1
P9(B)	17	19	9	10	1.0	105907	5	2	11	20	100	62378	1	2
P10(A)	36	55 ⁵	22	29	0.0	0.5829	2	8	25	45	0.1	0.6233	2	25
P11(A)	35	59	12	12	0.0	0.0	1	1	12	12	0.0	0.0	1	1

Itns number of iterations.

Evals number of function evaluations in the stepsize strategy.

c_0 initial penalty value.

c_* threshold penalty value beyond which c was not required to increase.

k_{c_*} iteration at which c_* was achieved.

$k_{\tau=1}$ iteration at which constant unit stepsize was attained.

¹converged to (0.00000, -0.00055); exact solution: (0, 0).

²converged to (0.00000, -0.00039); exact solution: (0, 0).

³converged to global solution.

⁴converged to local solution (except NAG for initial point C).

⁵converged to (0.00055, 0.00020); exact solution: (0, 0).

F¹ failed to converge in 5000 iterations.

F² converged to wrong solution: (-1.40540, 0.52428); correct solution: (0.64645, 1.35355).

F³ converged to wrong solution: (0.60199, 1.74439, -0.00543, 4.75425, 0.96904, 2.01743, 2.16165); correct solution: (2.33050, 1.95137, -0.47754, 4.36573, -0.62449, 1.03813, 1.59423).

better of these two solutions. A similar initialization experiment with E04VDF of NAG was not possible.

In addition to providing comparative evaluations in terms of number of iterations and function evaluations, we also demonstrate the attainment of unit stepsizes by the algorithm and the achievement of the threshold value for the penalty term. The

algorithm did not need to increase the penalty further beyond this threshold.

In all the results reported, a small initial choice for c_0 , the initial penalty value, was adequate. In a significant number of cases, the penalty parameter increase rule (3.7) was not activated and in all other cases the increase did occur only in the initial few iterations. Convergence to unit stepsizes was also achieved in all cases.

In conclusion, the numerical results indicate that the algorithm is very promising. The adaptive penalty parameter rule provides a useful means of enforcing feasibility while the algorithm is ensured of progress towards the solution. Problems 3(A), 4(C), and 7(B) cannot be solved by EO4DAF of NAG due to convergence failure or convergence to the wrong values. These are solved by the present algorithm at a convergence rate similar to other starting points of the same problem. A comparison shows that all problems except 4(D), 5(B), and 8(A) are solved faster than E04VDF of NAG.

Problem 1 (Charalambous and Bandler (1976); constraints added).

$$f_1(x) = x_1^2 + x_2^4; \quad f_2(x) = (2 - x_1)^2 + (2 - x_2)^2, \quad f_3(x) = 2 \exp(x_2 - x_1),$$

$$h_1(x) = -x_1^4 - 2x_2^2 - 1, \quad h_2(x) = 2x_1^3 - x_2^2 - 2.5,$$

$$\text{solution } x_* = (1.13904, 0.89956), \quad \text{min-max value: } 1.95222,$$

$$\text{initial points: (A) } x_0 = (1.0, -0.1), \quad \text{(B) } x_0 = (0.0, 0.0),$$

$$\text{(C) } x_0 = (2.0, 2.0), \quad \text{(D) } x_0 = (4.0, -4.0).$$

Problem 2 (Charalambous and Bandler (1976); constraints added).

$$f_1(x) = x_1^2 + x_2^2 + x_1x_2, \quad f_2(x) = -f_1(x), \quad f_3(x) = \sin(x_1);$$

$$f_4(x) = -f_3(x), \quad f_5(x) = \cos(x_2), \quad f_6(x) = -f_5(x),$$

$$h_1(x) = -x_1^4 - 2x_2^2 - 1, \quad h_2(x) = 2x_1^3 - x_2^2 - 2.5,$$

$$\text{solution } x_* = (-0.45330, 0.90659), \quad \text{min-max value: } 0.61643,$$

$$\text{initial points: (A) } x_0 = (3.0, 1.0), \quad \text{(B) } x_0 = (1.0, 3.0).$$

Problem 3 (Polak, Higgins, and Mayne, (1992, Problem WF); constraints added).

$$f_1(x) = \frac{1}{2} \left(x_1 + \frac{10x_1}{(x_1 + 0.1)} + 2x_2^2 \right), \quad f_2(x) = \frac{1}{2} \left(-x_1 + \frac{10x_1}{(x_1 + 0.1)} + 2x_2^2 \right),$$

$$f_3(x) = \frac{1}{2} \left(x_1 - \frac{10x_1}{(x_1 + 0.1)} - 2x_2^2 \right),$$

$$h_1(x) = x_1^2 + x_2^2 + x_1x_2, \quad h_2(x) = -x_1 + x_2^2,$$

$$\text{solution } x_* = (0.0, 0.0), \quad \text{min-max value: } 0.0,$$

$$\text{initial points: (A) } x_0 = (3.0, 1.0), \quad \text{(B) } x_0 = (50.0, 0.05).$$

Problem 4 (Conn (1979); constraints added).

$$f_1(x) = x_1^2 + x_2^2, \quad f_2(x) = (2 - x_1)^2 + (2 - x_2)^2, \quad f_3(x) = 2 \exp(x_2 - x_1),$$

$$h_1(x) = x_1 + x_2 - 2, \quad h_2(x) = -x_1^2 - x_2^2 + 2.25,$$

global solution $x_* = (1.35355, 0.64645)$, min-max value: 2.25,

local solution $x_* = (0.64645, 1.35355)$, min-max value: 4.05623,

initial points: (A) $x_0 = (2.1, 1.9)$, (B) $x_0 = (1.9, 2.1)$,

(C) $x_0 = (2.0, 4.0)$, (D) $x_0 = (4.0, 2.0)$.

Problem 5 (Asaadi (1973); Rosen–Suzuki problem).

$$f_1(x) = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4,$$

$$f_2(x) = f_1(x) + 10h_1(x), \quad f_3(x) = f_1(x) + 10h_2(x), \quad f_4(x) = f_1(x) + 10h_3(x),$$

$$h_1(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 - 8$$

$$h_2(x) = x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 - 10,$$

$$h_3(x) = x_1^2 + x_2^2 + x_3^2 + 2x_1 - x_2 - x_4 - 5,$$

solution $x_* = (0.0, 1.0, 2.0, -1.0)$, min-max value: -44.0

initial points: (A) $x_0 = (0.0, 0.0, 0.0, 0.0)$, (B) $x_0 = (0.0, 1.0, 1.0, 0.0)$;

(C) $x_0 = (2.0, 2.0, 5.0, 0.0)$, (D) $x_0 = (1.0, 3.0, 3.0, 1.0)$, (E) $x_0 = (-2.0, 1.0, 1.0, -2.0)$.

Problem 6 (Charalambous and Bandler (1976); constraints added).

$$f_1(x) = x_1^4 + x_2^2, \quad f_2(x) = (2 - x_1)^2 + (2 - x_2)^2, \quad f_3(x) = 2 \exp(x_2 - x_1),$$

$$h_1(x) = x_1^2 - x_2^2, \quad h_2(x) = -2x_1^3 - x_2^2,$$

solution $x_* = (1.0, 1.0)$, min-max value: 2.0,

initial points: (A) $x_0 = (0.0, 1.0)$, (B) $x_0 = (3.0, 1.0)$.

Problem 7 (Asaadi (1973, Wong 1)).

$$f_1(x) = (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2$$

$$+ 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7,$$

$$f_2(x) = f_1(x) + 10h_1(x), \quad f_3(x) = f_1(x) + 10h_2(x),$$

$$f_4(x) = f_1(x) + 10h_3(x), \quad f_5(x) = f_1(x) + 10h_4(x),$$

$$h_1(x) = 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 - 127,$$

$$h_2(x) = 7x_1 + 3x_2 + 10x_2^2 + x_4 - x_5 - 282,$$

$$h_3(x) = 23x_1 + x_2^2 + 6x_6^2 - 8x_7 - 196,$$

$$h_4(x) = 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7,$$

solution $x_* = (2.33050, 1.95137, -0.47754, 4.36573, -0.62449, 1.03813, 1.59423)$,

min-max value: 680.63006,

initial points: (A) $x_0 = (1.0, 2.0, 0.0, 4.0, 0.0, 1.0, 1.0)$,

(B) $x_0 = (3.0, 3.0, 0.0, 5.0, 1.0, 3.0, 0.0)$.

Problem 8 (Hald and Madsen (1981); constraints added).

$$\begin{aligned} f_1(x) &= 10(x_2 - x_1^2), \quad f_2(x) = -f_1(x), \quad f_3(x) = 1 - x_1, \quad f_4(x) = -f_3(x), \\ h_1(x) &= 100x_1^2 + x_2^2 - 101, \quad h_2(x) = 80x_1^2 - x_2^2 - 79, \\ \text{solution } x_* &= (1.0, 1.0), \quad \text{min-max value: } 0.0, \\ \text{initial point: (A) } x_0 &= (-1.2, 1.0). \end{aligned}$$

Problem 9 (Polak, Mayne, and Higgins (1991); constraints added).

$$\begin{aligned} f_1(x) &= \exp\left(\frac{x_1^2}{1000} + (x_2 - 1)^2\right), \quad f_2(x) = \exp\left(\frac{x_1^2}{1000} + (x_2 + 1)^2\right), \\ h_1(x) &= \frac{x_1^2}{1000} + x_2^2 + x_1x_2, \quad h_2(x) = -x_1 + x_2^2, \\ \text{solution } x_* &= (0.0, 0.0), \quad \text{min-max value: } 2.71828, \\ \text{initial points: (A) } x_0 &= (50.0, 0.05), \quad \text{(B) } x_0 = (1.0, 1.1). \end{aligned}$$

Problem 10 (Polak, Mayne, and Higgins (1992, Problem SPIRAL); constraints added).

$$\begin{aligned} f_1(x) &= \left(x_1 - \sqrt{x_1^2 + x_2^2} \cos\left(\sqrt{x_1^2 + x_2^2}\right)\right)^2 + 0.005(x_1^2 + x_2^2), \\ f_2(x) &= \left(x_2 - \sqrt{x_1^2 + x_2^2} \sin\left(\sqrt{x_1^2 + x_2^2}\right)\right)^2 + 0.005(x_1^2 + x_2^2), \\ h_1(x) &= x_1^2 + x_2^2 - 3, \quad h_2(x) = -x_1 - x_2 - 2, \\ \text{optimal solution: } x_* &= (0, 0), \quad \text{min-max value: } 0.0, \\ \text{initial point: (A) } x_0 &= (1.41831, -4.79462). \end{aligned}$$

Problem 11 (Asaadi (1973, Wong 2)).

$$\begin{aligned} f_1(x) &= x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 \\ &\quad + 4(x_4 - 5)^2 + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 \\ &\quad + 7(x_8 - 11) + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45, \\ f_2(x) &= f_1(x) + 20h_1, \quad f_3(x) = f_1(x) + 20h_2, \\ f_4(x) &= f_1(x) + 20h_3, \quad f_5(x) = f_1(x) + 20h_4, \\ f_6(x) &= f_1(x) + 20h_5, \quad f_7(x) = f_1(x) + 20h_6, \\ f_8(x) &= f_1(x) + 20h_7, \quad f_9(x) = f_1(x) + 20h_8, \end{aligned}$$

$$h_1(x) = 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120,$$

$$h_2(x) = 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 4,$$

$$h_3(x) = \frac{1}{2}(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30,$$

$$h_4(x) = x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6,$$

$$h_5(x) = 4x_1 + 5x_2 - 3x_7 + 9x_8 - 105,$$

$$h_6(x) = 10x_1 - 8x_2 - 17x_7 + 2x_8,$$

$$h_7(x) = -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10},$$

$$h_8(x) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12,$$

solution:⁵

$$x_* = (0.87473, 1.97024, 6.32300, 7.84599, 0.09147, 3.64972,$$

$$0.81747, 1.45581, 6.82687, 9.53850),$$

$$\text{min-max value: } 108.01669,$$

$$\text{initial point: (A) } x_0 = (2.0, 3.0, 5.0, 5.0, 1.0, 2.0, 7.0, 3.0, 6.0, 10.0).$$

REFERENCES

- K. J. ARROW, F. J. GOULD, AND S. M. HOWE (1973), *General saddle point result for constrained optimization*, Math. Programming, 5, pp. 225–234.
- K. J. ARROW AND R. M. SOLOW (1958), *Gradient methods for constrained maxima, with weakened assumptions*, in Studies in Linear and Nonlinear Programming, K. Arrow, L. Hurwicz, and H. Uzawa, eds., Stanford University Press, Stanford, CA, pp. 166–176.
- J. ASAADI (1973), *A computational comparison of nonlinear programs*, Math. Programming, 4, pp. 144–154.
- M. C. B. BIGGS (1974), *The Development of a Class of Constrained Minimization Algorithms and their Application to the Problem of Power Scheduling*, Ph.D. thesis, Department of Computer Science, Queen Mary College, University of London, London, UK.
- C. CHARALAMBOUS AND J. W. BANDLER (1976), *Nonlinear minimax optimization as a sequence of least pth optimization with finite values of p*, Internat. J. Systems Sci., 7, pp. 377–394.
- C. CHARALAMBOUS AND A. R. CONN (1978), *An efficient algorithm to solve the min-max problem directly*, SIAM J. Numer. Anal., 15, pp. 162–187.
- G. C. CHOW (1979), *Effective use of econometric models in macroeconomic policy formulation*, in Optimal Control for Econometric Models, S. Holly, B. Rustem, and M. Zarrop, eds., Macmillan, London, pp. 31–39.
- T. F. COLEMAN (1978), *A note on “New algorithms for constrained minimax optimization,”* Math. Programming, 15, pp. 239–242.
- A. R. CONN (1979), *An Efficient Second Order Method to Solve the Constrained Minmax Problem*, Report, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada.

⁵The solution reported by Asaadi (1973) does not correspond to the problem reported in that paper. As references to Wong 2 in the literature cite Asaadi (1973), we have adopted this formulation.

- A. R. CONN AND Y. LI (1992), *A structure exploiting algorithm for nonlinear minimax problems*, SIAM J. Optim., 2, pp. 242–263.
- V. F. DEMYANOV AND V. N. MALOZEMOV (1974), *Introduction to Minmax*, John Wiley, New York.
- V. F. DEMYANOV AND A. B. PEVNYI (1972), *Some estimation in minmax problems*, Kibernetika, 1, pp. 107–112 (in Russian).
- S. R. K. DUTTA AND M. VIDYASAGAR (1977), *New algorithms for constrained minmax optimization*, Math. Programming, 13, pp. 149–155.
- J. FUHRER AND J. HALTMAIER (1986), *Minimum Variance Pooling of Forecasts at Different Levels of Aggregation*, Special Studies Paper 208, Federal Reserve Board, Washington, D.C.
- P. E. GILL, W. MURRAY, AND M. H. WRIGHT (1981), *Practical Optimization*, Academic Press, New York.
- C. GRANGER AND P. NEWBOLD (1977), *Forecasting Economic Time Series*, Academic Press, New York.
- J. H. HALD AND K. MADSEN (1981), *Combined LP and quasi-Newton methods for minimax optimization*, Math. Programming, 20, pp. 49–62.
- S. P. HAN (1981), *Variable metric methods for minimizing a class of nondifferentiable functions*, Math. Programming, 20, pp. 1–13.
- M. J. LAWRENCE, R. H. EDMUNSON, AND M. J. O'CONNOR (1986), *The accuracy of combining judgmental and statistical forecasts*, Management Science, 32, pp. 1521–1532.
- S. MAKRIDAKIS AND R. WINKLER (1983), *Averages of forecasts: Some empirical results*, Management Science, 29, pp. 987–996.
- W. MURRAY AND M. L. OVERTON (1980), *A projected Lagrangian algorithm for nonlinear minmax optimization*, SIAM J. Sci. Statist. Comput., 1, pp. 345–370.
- E. POLAK, J. E. HIGGINS, AND D. Q. MAYNE (1992), *A barrier function method for minimax problems*, Math. Programming, 54, pp. 155–176.
- E. POLAK, D. Q. MAYNE, AND J. E. HIGGINS (1991), *A superlinearly convergent min-max algorithm for min-max problems*, J. Optim. Theory Appl., 69, pp. 407–439.
- E. POLAK AND A. L. TITS (1980), *A globally convergent, implementable multiplier method with automatic penalty limitation*, Appl. Math. Optim., 6, pp. 335–360.
- M. J. D. POWELL (1978), *A fast algorithm for nonlinearly constrained optimization problems*, in Numerical Analysis, Dundee, 1977, Lecture Notes in Mathematics 630, G. A. Watson, ed., Springer-Verlag, Berlin, pp. 144–157.
- R. T. ROCKAFELLAR (1973), *A dual approach to solving nonlinear programming problems by unconstrained minimization*, Math. Programming, 5, pp. 354–373.
- B. RUSTEM (1987), *Methods for the simultaneous use of multiple models in optimal policy design*, in Developments in Control Theory for Economic Analysis, C. Carraro and D. Sartore, eds., Martinus Nijhoff-Kluwer Publishers, Dordrecht, The Netherlands, pp. 157–186.
- B. RUSTEM (1992), *A superlinearly convergent min-max algorithm for rival models of the same economic system*, Math. Programming, 53, pp. 279–295.
- B. RUSTEM (1993), *Equality and inequality constrained optimization algorithms with convergent stepsizes*, J. Optim. Theory Appl., 76, pp. 429–453.
- R. A. TAPIA (1986), *On Secant Updates for Use in General Constrained Optimization*, Technical report 84-3, Department of Mathematical Sciences, Rice University, Houston, TX.
- R. S. WOMERSLEY AND R. FLETCHER (1986), *An algorithm for composite nonsmooth optimization problems*, J. Optim. Theory Appl., 48, pp. 493–523.

ERRATUM: AN SQP ALGORITHM FOR FINELY DISCRETIZED CONTINUOUS MINIMAX PROBLEMS AND OTHER MINIMAX PROBLEMS WITH MANY OBJECTIVE FUNCTIONS*

JIAN L. ZHOU[†] AND ANDRÉ L. TITS[‡]

Abstract. An error is pointed out in the local convergence proof in the quoted paper [J. L. Zhou and A. L. Tits, *SIAM J. Optim.*, 6 (1996), pp. 461–487]. A correct proof is given.

Key words. continuous minimax, semi-infinite programming, many constraints, sequential quadratic programming, discretization, global convergence

AMS subject classifications. 49M07, 49M37, 49M39, 65K05, 90C06, 90C30, 90C34

PII. S1052623496313659

The proof of Lemma 3.14 in [2] is incorrect. Namely, proving the claim in the second sentence of the proof does not “complete the proof” as stated. To see this, note that the mathematical induction argument hinted at in that sentence merely proves that, for the infinite index set K whose existence is established by Lemma 3.12, given any integer i , there exists k_i such that $\Omega_{\max}(x^*) \subseteq \Omega_{k+i}^b$ for all $k \in K$, $k \geq k_i$. Since $\{k_i : i = 1, 2, \dots\}$ is not shown to be bounded, it does not follow that $\Omega_{\max}(x^*) \subseteq \Omega_k^b$ for all k large enough.

Nevertheless, the convergence results claimed in that section of [2] do hold true. A correct proof is obtained if Lemmas 3.13, 3.14, and 3.15 in [2] are replaced with Lemmas 3.13', 3.14', and 3.15' given below. (The statements of Lemmas 3.14' and 3.15' are identical to those of Lemmas 3.14 and 3.15, respectively, but the proofs are different.) The implicit assumptions required for the original Lemmas 3.13–3.15 are still assumed to hold.

LEMMA 3.13'. Let $\sigma_1, \sigma_2 > 0$ be given, and let

$$\mathcal{H} = \{H = H^T : \sigma_1 \|d\|^2 \leq \langle d, Hd \rangle \leq \sigma_2 \|d\|^2 \quad \forall d \in \mathbb{R}^n\}.$$

Then, for every $\epsilon > 0$, there exists $\delta > 0$ such that for every x with $\|x - x^*\| < \delta$, every $H \in \mathcal{H}$, and every $\hat{\Omega} \subseteq \Omega$ with $\Omega_{\max}(x^*) \subseteq \hat{\Omega}$, all $\omega \in \Omega_{\max}(x^*)$ are binding for $QP(x, H, \hat{\Omega})$ and the unique KKT point d of $QP(x, H, \hat{\Omega})$ satisfies $\|d\| < \epsilon$.

Proof. Given x with $\|x - x^*\| < \delta$, $H \in \mathcal{H}$, and $\hat{\Omega} \subseteq \Omega$ with $\Omega_{\max}(x^*) \subseteq \hat{\Omega}$, let $d(x, H, \hat{\Omega})$ be the unique KKT point of $QP(x, H, \hat{\Omega})$. Then Lemma 3.2 implies that $d(x^*, H, \hat{\Omega}) = 0$ for all $H \in \mathcal{H}$. Since $\mathcal{H}$ is compact, in view of Assumptions 4 and 5, it follows from a classical result of Robinson [1, Theorem 2.1] that, given $\epsilon > 0$, there exists $\delta_{\hat{\Omega}} > 0$ such that, for all x with $\|x - x^*\| < \delta_{\hat{\Omega}}$ and all $H \in \mathcal{H}$, all $\omega \in \Omega_{\max}(x^*)$ are binding for $QP(x, H, \hat{\Omega})$ and $\|d(x, H, \hat{\Omega})\| < \epsilon$. That $\delta_{\hat{\Omega}} > 0$ can be chosen independent of $\hat{\Omega}$ follows from the finiteness of Ω . $\square$

LEMMA 3.14'. For k large enough, $\Omega_{\max}(x^*) \subseteq \Omega_k^b$.

*Received by the editors December 10, 1996; accepted for publication September 15, 1997. This research was supported in part by NSF's Engineering Research Centers Program NSF-CDR-88-03012, by NSF grant DMC-88-15996, and by a grant from the Westinghouse Corporation.

<http://www.siam.org/journals/siopt/8-1/31365.html>

[†]Fannie Mae, Washington, DC 20016 (sxujxz@fnma.com).

[‡]Institute for Systems Research and Electrical Engineering Department, University of Maryland, College Park, MD 20742 (andre@src.umd.edu).

Proof. Let $\sigma_1, \sigma_2 > 0$ be as given by Assumption 3, and let $\delta > 0$ be as given by Lemma 3.13' (for an arbitrary $\epsilon > 0$). Since $x_k \rightarrow x^*$ as $k \rightarrow \infty$ (Proposition 3.11), it follows from Lemma 3.12 that there exists $\hat{k}$ such that $\|x_k - x^*\| < \delta$ for all $k \geq \hat{k}$ and $\Omega_{\max}(x^*) \subseteq \Omega_{\hat{k}}$. That $\Omega_{\max}(x^*) \subseteq \Omega_k$ for all $k \geq \hat{k}$ can now be proved by mathematical induction. Indeed, let $k \geq \hat{k}$ and suppose $\Omega_{\max}(x^*) \subseteq \Omega_k$. It follows from Lemma 3.13' and Assumption 3 that $\Omega_{\max}(x^*) \subseteq \Omega_k^b$ and, since, by construction, $\Omega_k^b \subseteq \Omega_{k+1}$ for all k , it follows that $\Omega_{\max}(x^*) \subseteq \Omega_{k+1}$. That $\Omega_{\max}(x^*) \subseteq \Omega_k^b$ for all $k \geq \hat{k}$ now follows directly from Lemma 3.13' and Assumption 3. $\square$

LEMMA 3.15'. *The entire sequence $\{d_k\}$ converges to zero.*

Proof. The proof of this lemma follows directly from Lemmas 3.13' and 3.14', Assumption 3, and the fact that, by construction, $\Omega_k^b \subseteq \Omega_{k+1}$ for all k . $\square$

Acknowledgment. The authors wish to thank Craig T. Lawrence for uncovering the error in the proof of Lemma 3.14 in [2].

REFERENCES

- [1] S. M. ROBINSON, *Perturbed Kuhn–Tucker points and rates of convergence for a class of nonlinear-programming algorithms*, Math. Programming, 7 (1974), pp. 1–16.
- [2] J. L. ZHOU AND A. L. TITS, *An SQP algorithm for finely discretized continuous minimax problems and other minimax problems with many objective functions*, SIAM J. Optim., 6 (1996), pp. 461–487.

TILT STABILITY OF A LOCAL MINIMUM*

R. A. POLIQUIN[†] AND R. T. ROCKAFELLAR[‡]

Abstract. The behavior of a minimizing point when an objective function is tilted by adding a small linear term is studied from the perspective of second-order conditions for local optimality. The classical condition of a positive-definite Hessian in smooth problems without constraints is found to have an exact counterpart much more broadly in the positivity of a certain generalized Hessian mapping. This fully characterizes the case where tilt perturbations cause the minimizing point to shift in a Lipschitzian manner.

Key words. variational analysis, second-order optimality, tilt stability, perturbations, sensitivity, prox-regularity, amenable functions, generalized Hessians, coderivatives, monotone mappings

AMS subject classifications. Primary, 49A52, 58C06, 58C20; Secondary, 90C30

PII. S1052623496309296

1. Introduction. Much of the theory of optimality conditions has proceeded from the fact that, for a $\mathcal{C}^2$ function f on $\mathbb{R}^n$, a necessary condition for a local minimum at a point $\bar{x}$ is the vanishing of the gradient $\nabla f(\bar{x})$ and the positive-semidefiniteness of the Hessian $\nabla^2 f(\bar{x})$, whereas a sufficient condition is the vanishing of $\nabla f(\bar{x})$ and the positive-definiteness of $\nabla^2 f(\bar{x})$. Analogous patterns have been found for problems with $\mathcal{C}^2$ constraints and also for problems exhibiting nonsmoothness. The goal has been to come up with necessary conditions that become sufficient with only a slight degree of strengthening. But is this the best paradigm for modern purposes?

Optimization no longer revolves around making lists of solution candidates to be checked out one by one, if it ever did. The role of optimality conditions is seen rather in the justification of numerical algorithms, in particular their stopping criteria, convergence properties, and robustness. From that angle, the goal of theory could be different. Instead of focusing on the threshold between necessity and sufficiency, one might more profitably try to characterize the stronger manifestations of optimality that support computational work. Indeed, this idea has motivated much of the effort that has gone into parametric optimization—the study of how local solutions to a problem may react to shifts in data.

Here we take up this theme more narrowly, but in other respects more broadly, than before. We investigate just one fundamental mode of perturbation, but do so in a very general setting.

Any problem of optimization in n variables can be expressed as one of minimizing a function f over all of $\mathbb{R}^n$, as long as f is allowed to have values in the extended-real-line $\overline{\mathbb{R}}$ as a means of representing constraints through infinite penalties. If the aim is to minimize a function f_0 subject to a system of constraints, this is captured by defining $f(x)$ to be $f_0(x)$ when x belongs to the feasible set C , but letting $f(x) = \infty$ when

*Received by the editors September 11, 1996; accepted for publication (in revised form) March 25, 1997.

<http://www.siam.org/journals/siopt/8-2/30929.html>

[†]Department of Mathematical Sciences, University of Alberta, Edmonton, AB, T6G 2G1, Canada (rene.poliquin@math.ualberta.ca). The research of this author was supported by the Natural Sciences and Engineering Research Council of Canada under grant OGP41983. This paper was written in part while the author was visiting the Université de Perpignan.

[‡]Department of Mathematics, University of Washington, Seattle, WA 98195 (rtr@math.washington.edu). The research of this author was supported in part by the National Science Foundation under grant DMS-9500957.

$x \notin C$. Keeping for now to such a level of abstraction, with the idea of introducing various details of problem structure later, we are able to formulate succinctly the property we propose to analyze.

DEFINITION 1.1. *A point $\bar{x}$ will be said to give a tilt-stable local minimum of the function $f : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$ if $f(\bar{x})$ is finite and there exists $\delta > 0$ such that the mapping*

$$M : v \mapsto \operatorname{argmin}_{|x - \bar{x}| \leq \delta} \{ f(x) - f(\bar{x}) - \langle v, x - \bar{x} \rangle \}$$

is single-valued and Lipschitzian on some neighborhood of $v = 0$ with $M(0) = \bar{x}$.

The Lipschitzian requirement means the existence of a constant $\kappa \geq 0$ for which $|M(v'') - M(v')| \leq \kappa|v'' - v'|$ when $|v'|$ and $|v''|$ are small enough. The subtraction from f of the affine function $l(x) = f(\bar{x}) + \langle v, x - \bar{x} \rangle$, which agrees with f at $\bar{x}$, serves to tilt the objective in one direction or another, and the requirement is that such a perturbation, if sufficiently small, should not affect the local solution disproportionately to the size of v or threaten its uniqueness. The connection between this notion of tilt stability and the classical picture of local optimality is as follows.

PROPOSITION 1.2. *For a C^2 function f having $\nabla f(\bar{x}) = 0$, the point $\bar{x}$ gives a tilt-stable local minimum of f if and only if $\nabla^2 f(\bar{x})$ is positive-definite.*

Proof. Sufficiency. We have $\nabla^2 f(x)$ positive-definite for all x in a convex neighborhood X of $\bar{x}$. In particular, f is strictly convex on X , so that for $\tilde{x} \in X$ and $\tilde{v} = \nabla f(\tilde{x})$ we have

$$f(x) > f(\tilde{x}) + \langle \tilde{v}, x - \tilde{x} \rangle \text{ for all } x \in X, x \neq \tilde{x}.$$

Then for $\delta > 0$ small enough the mapping M in Definition 1.1 is single-valued on $\nabla f(X)$ with $M(0) = \bar{x}$, and it agrees on that set with the inverse of the restriction of ∇f to X . The standard implicit mapping theorem tells us that M is C^1 around 0. Hence M is Lipschitzian on some neighborhood of 0.

Necessity. From $M(0) = \bar{x}$ we have $\nabla^2 f(\bar{x})$ positive-semidefinite. Suppose that $\nabla^2 f(\bar{x})$ is not positive-definite and thus is singular. There is a vector $\bar{w} \notin \{\nabla^2 f(\bar{x})w \mid w \in \mathbb{R}^n\}$. For $\tau > 0$ so small that $\tau\bar{w}$ lies in a neighborhood where M is single-valued and Lipschitzian with constant κ , we have $|M(\tau\bar{w}) - M(0)| \leq \kappa\tau|\bar{w}|$. Let $x^\nu = M(\tau^\nu\bar{w})$ for any sequence of values $\tau^\nu \downarrow 0$ (with superscript $\nu \rightarrow \infty$). The sequence of vectors $w^\nu = [x^\nu - \bar{x}]/\tau^\nu$ is bounded in norm by $\kappa|\bar{w}|$, so, by passing to a subsequence if necessary, we can assume that it converges to some $\tilde{w}$. Because $\nabla f(x^\nu) = \tau^\nu\bar{w}$ and $\nabla f(\bar{x}) = 0$, we have $\bar{w} = [\nabla f(\bar{x} + \tau^\nu w^\nu) - \nabla f(\bar{x})]/\tau^\nu \rightarrow \nabla^2 f(\bar{x})\tilde{w}$, hence $\bar{w} = \nabla^2 f(\bar{x})\tilde{w}$, which is impossible. $\square$

Can some analogous characterization of tilt stability be given for functions f of a wider class? This question has not previously been asked. In looking for an answer in the absence of f being differentiable, we must appeal to optimality conditions based on subgradients.

For any function $f : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$ and any point $\bar{x}$ with $f(\bar{x})$ finite, a vector $\bar{v}$ is a *proximal subgradient* of f at $\bar{x}$ if there exist $\delta > 0$ and $r \geq 0$ such that

$$f(x) \geq f(\bar{x}) + \langle \bar{v}, x - \bar{x} \rangle - \frac{r}{2}|x - \bar{x}|^2 \text{ when } |x - \bar{x}| \leq \delta.$$

It is a *subgradient* (in general), written $\bar{v} \in \partial f(\bar{x})$, if there is a sequence of points $x^\nu \rightarrow \bar{x}$ with $f(x^\nu) \rightarrow f(\bar{x})$ at which there exist proximal subgradients v^ν such that $v^\nu \rightarrow \bar{v}$.

When f is convex, $\partial f(\bar{x})$ is the usual subgradient set of convex analysis. When f is smooth, $\partial f(\bar{x})$ is the singleton $\{\nabla f(\bar{x})\}$. For the indicator function δ_C associated

with a set C (which vanishes on C but is ∞ outside of C), the subgradients v at $\bar{x}$ are the *normal vectors* to C at $\bar{x}$; the set $\partial\delta_C(\bar{x})$ is the *normal cone* to C at $\bar{x}$, denoted by $N_C(\bar{x})$. The book [1] provides details on this topic and other aspects of variational analysis that will be important in what follows.

Two properties of f that relate to the set-valued subgradient mapping $x \mapsto \partial f(x)$ and its graph

$$\text{gph } \partial f = \{ (x, v) \mid v \in \partial f(x) \}$$

will be important. We say that f is *subdifferentially continuous* at $\bar{x}$ for a vector $\bar{v} \in \partial f(\bar{x})$ if the function $(x, v) \mapsto f(x)$ is continuous relative to $\text{gph } \partial f$ at $(\bar{x}, \bar{v})$, or in other words, if for every $\delta > 0$ there exists $\epsilon > 0$ such that $|f(x) - f(\bar{x})| < \delta$ when $v \in \partial f(x)$ with $|x - \bar{x}| < \epsilon$ and $|v - \bar{v}| < \epsilon$. On the other hand, f is *prox-regular* at $\bar{x}$ for $\bar{v}$ if its epigraph is closed relative to a neighborhood of $(\bar{x}, f(\bar{x}))$ and there exist $\epsilon > 0$ and $r > 0$ such that

$$f(x') > f(x) + \langle v, x' - x \rangle - \frac{r}{2} |x' - x|^2 \text{ for } x' \neq x \text{ when} \\ |x' - \bar{x}| < \epsilon, |x - \bar{x}| < \epsilon, |f(x) - f(\bar{x})| < \epsilon, |v - \bar{v}| < \epsilon, v \in \partial f(x).$$

Prox-regularity thus requires, in a certain uniform local sense with respect to $\bar{x}$ and $\bar{v}$, that all subgradients $v \in \partial f(x)$ actually be proximal subgradients.

The concepts of prox-regularity and subdifferential continuity were introduced in [2] and studied further in [3] and [4]. It was shown in those works that the class of functions exhibiting these properties is very large. It includes, in addition to all $\mathcal{C}^2$ functions f , all lower semicontinuous (lsc) proper convex functions f , as well as many of the functions that typically might be encountered in finite-dimensional optimization. For instance, if $f = f_0 + \delta_C$ for a function f_0 that is $\mathcal{C}^2$, or expressible as a max of finitely many $\mathcal{C}^2$ functions, and a set C is specified by a finite family of equality and/or inequality constraints on $\mathcal{C}^2$ functions, then at any point $\bar{x}$ where the constraint system satisfies the Mangasarian–Fromovitz constraint qualification the function f is both prox-regular and subdifferentially continuous for every $\bar{v} \in \partial f(\bar{x})$.

More generally, the class of prox-regular, subdifferentially continuous functions includes all *strongly amenable* functions, which are defined as being representable locally by the composition of an lsc proper convex function with a $\mathcal{C}^2$ mapping under a constraint qualification on the domain of the convex function and the range of the linearization of the mapping; cf. [5] and also [6], [7], [8]. Such functions are omnipresent in applications, and they have a basic role both theoretically and computationally. For more on the use of composite function formats in optimization, see [4], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19].

The first-order condition $0 \in \partial f(\bar{x})$ is always necessary for f to have a local minimum at $\bar{x}$. Second-order conditions have been developed in various forms, but here we turn to a new form in terms of the generalized Hessian mapping introduced by Mordukhovich [20], [21]. For any point $\bar{x}$ and any subgradient $\bar{v} \in \partial f(\bar{x})$, define $\partial^2 f(\bar{x} | \bar{v}) : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ by

$$\partial^2 f(\bar{x} | \bar{v}) : w \mapsto \{ z \mid (z, -w) \in N_{\text{gph } \partial f}(\bar{x}, \bar{v}) \}.$$

When f is $\mathcal{C}^2$ and $\bar{v} = \nabla f(\bar{x})$, $\partial^2 f(\bar{x} | \bar{v})$ reduces to the linear mapping $w \mapsto \nabla^2 f(\bar{x})w$.

In situations where f is given in terms of other functions, it is often possible to draw upon the growing calculus of “coderivatives” to compute the generalized Hessian

$\partial^2 f(\bar{x}|\bar{v})$. For any set-valued mapping $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^m$, the *coderivative* at $\bar{x}$ for an element $\bar{v} \in T(\bar{x})$ is the mapping $D^*T(\bar{x}|\bar{v}) : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$ whose graph is obtained from the normal cone $N_{\text{gph } T}(\bar{x}, \bar{v})$ by

$$z \in D^*T(\bar{x}|\bar{v})(w) \iff (z, -w) \in N_{\text{gph } T}(\bar{x}, \bar{v}).$$

In such terms, one has $\partial^2 f(\bar{x}|\bar{v}) = D^*[\partial f](\bar{x}|\bar{v})$. For more on coderivatives, their history and calculus see [1], [22], and [23].

Our main result can now be stated. It will be proved in section 3, following the development of a monotonicity property in section 2. Application to the constrained minimization of a smooth function will be made in section 4.

THEOREM 1.3. *For a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ having $0 \in \partial f(\bar{x})$ and such that f is both prox-regular and subdifferentially continuous at $\bar{x}$ for $\bar{v} = 0$, the following are equivalent and imply the existence of $\delta > 0$ such that, for all v in some neighborhood of 0, the mapping M in Definition 1.1 has $M(v)$ as the unique $x \in (\partial f)^{-1}(v)$ with $|x - \bar{x}| \leq \delta$.*

- (a) *The point $\bar{x}$ gives a tilt-stable local minimum of f .*
- (b) *The mapping $\partial^2 f(\bar{x}|0)$ is positive-definite in the sense that*

$$\langle z, w \rangle > 0 \quad \text{whenever} \quad z \in \partial^2 f(\bar{x}|0)(w), \quad w \neq 0.$$

- (c) *There exist neighborhoods X of $\bar{x}$ and V of 0 such that the mapping $x \in X \mapsto \partial f(x) \cap V$ is strongly monotone and, locally around $(\bar{x}, 0)$, is maximal monotone.*
- (d) *There is a proper, lsc, strongly convex function h on $\mathbb{R}^n$ along with neighborhoods X of $\bar{x}$ and V of 0 such that h is finite on X with $h(\bar{x}) = f(\bar{x})$ and $0 \in \partial h(\bar{x})$, and furthermore $\text{gph } \partial f \cap [X \times V] = \text{gph } \partial h \cap [X \times V]$.*

To understand (c), recall that a (generally set-valued) mapping $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is *monotone* if $\langle v' - v, x' - x \rangle \geq 0$ whenever $v \in T(x)$ and $v' \in T(x')$. It is *strongly monotone* when $T - \lambda I$ is monotone for some $\lambda > 0$. It is *maximal monotone*, locally around $(\bar{x}, \bar{v})$, if there exist neighborhoods X_0 of $\bar{x}$ and V_0 of $\bar{v}$ such that every monotone mapping $T' : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ with $\text{gph } T' \supset \text{gph } T \cap [X_0 \times V_0]$ has $\text{gph } T' \cap [X_0 \times V_0] = \text{gph } T \cap [X_0 \times V_0]$.

Observe in (d) that h need not agree with f on a neighborhood of $\bar{x}$. In contrast to the smooth case, tilt stability and the positive-definite generalized Hessian do not entail the local convexity of f around $\bar{x}$. An example is furnished by the function $f(x) = |1 - x^2|$ on $\mathbb{R}$. This has a tilt-stable minimum at $\bar{x} = 1$ and at $\bar{x} = -1$, although it is concave on the interval $(-1, 1)$.

2. Coderivatives of monotone mappings. The argument for our characterization of tilt stability will rely on certain monotonicity properties of the subgradient mappings associated with prox-regular functions. For this purpose we must first develop a fact about coderivatives in the presence of monotonicity.

THEOREM 2.1. *If a mapping $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is maximal monotone, one has for every pair $(\bar{x}, \bar{v}) \in \text{gph } T$ that*

$$\langle x', v' \rangle \geq 0 \quad \text{when } v' \in D^*T(\bar{x}|\bar{v})(x').$$

The verification of this fact will utilize the following connection between monotone mappings and nonexpansive mappings. A mapping $S : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is *nonexpansive* if $w_i \in S(z_i)$ for $i = 0, 1$ implies $|w_1 - w_0| \leq |z_0 - z_1|$. On the set $\text{dom } S = \{z \mid S(z) \neq \emptyset\}$, such a mapping must actually be single-valued.

PROPOSITION 2.2. *The one-to-one linear transformation $J : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^n \times \mathbb{R}^n$ with $J(x, v) = (v + x, v - x)$ induces a one-to-one correspondence between mappings $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ and mappings $S : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ through*

$$\text{gph } S = J(\text{gph } T), \quad \text{gph } T = J^{-1}(\text{gph } S),$$

in which T is monotone if and only if S is nonexpansive. Moreover, T is maximal monotone if and only if $\text{dom } S = \mathbb{R}^n$, or in other words, S is a single-valued mapping from all of $\mathbb{R}^n$ into itself that is globally Lipschitzian with constant 1.

Proof. See [1, Def. 12.11 and Thm. 12.12]. $\square$

Also to be used is the well-known scalarization formula for coderivatives of locally Lipschitzian mappings.

PROPOSITION 2.3. *For a single-valued, locally Lipschitzian mapping $S : \mathbb{R}^n \rightarrow \mathbb{R}^m$, any pair $(\bar{z}, \bar{w}) \in \text{gph } S$, and any vector $y \in \mathbb{R}^m$, one has $D^*S(\bar{z}|\bar{w})(y) = \partial(yS)(\bar{z})$ for the locally Lipschitzian function $yS : \mathbb{R}^n \rightarrow \mathbb{R}$ given by $(yS)(z) = \langle y, S(z) \rangle$.*

Proof. See [24] or [1, Prop. 9.24(b)]. $\square$

Proof of Theorem 2.1. Let S correspond to T as in Proposition 2.2, with the pair $(\bar{z}, \bar{w}) \in \text{gph } S$ corresponding to the pair $(\bar{x}, \bar{v}) \in \text{gph } T$. The linear transformation between the graph spaces induces a correspondence between normal cones to the graphs in question, which comes out as the property that

$$(v', -x') \in N_{\text{gph } T}(\bar{x}, \bar{v}) \iff (v' - x', -v' - x') \in N_{\text{gph } S}(\bar{z}, \bar{w}),$$

or equivalently

$$v' \in D^*T(\bar{x}|\bar{v})(x') \iff v' - x' \in D^*S(\bar{z}|\bar{w})(v' + x').$$

By Proposition 2.2, S is globally Lipschitzian with constant 1. Therefore, we have for any y that the mapping yS is globally Lipschitzian with constant $|y|$, so that for all $u \in \partial(yS)(\bar{z})$ we have $|u| \leq |y|$. Applying Proposition 2.3, we see that $|u| \leq |y|$ whenever $u \in D^*S(\bar{z}|\bar{w})(y)$. Applying this to $u = v' - x'$ and $y = v' + x'$ in the case of an arbitrary pair (x', v') with $v' \in D^*T(\bar{x}|\bar{v})(x')$, we obtain $|v' - x'| \leq |v' + x'|$. Hence $0 \leq |v' + x'|^2 - |v' - x'|^2 = 4\langle v', x' \rangle$, and we conclude that $\langle v', x' \rangle \geq 0$, as claimed. $\square$

3. Proof of the main result. In proving Theorem 1.3, we can focus on the case where $\bar{x} = 0$ and $f(\bar{x}) = 0$. These assumptions entail no real restriction and merely constitute a shift of variables. We may then assume also without loss of generality that there exists $r > 0$ with

$$(3.1) \quad f(x) > -\frac{r}{2}|x|^2 \text{ for all } x \neq 0.$$

Locally, this inequality comes out of the prox-regularity of f and the definition of a proximal subgradient. It is made into a global inequality by adding to f the indicator of a sufficiently small ball centered at 0, which is an operation having no effect on the property of tilt stability of f at $\bar{x} = 0$.

Recall that a mapping $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is *hypomonotone* around a pair $(x, v) \in \text{gph } T$ if $T + \alpha I$ is monotone near (x, v) for some $\alpha \geq 0$; cf. [25].

LEMMA 3.1. *Let $T_i : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ for $i = 1, 2$, with $(0, 0) \in \text{gph } T_1 \subset \text{gph } T_2$. Suppose T_1 is locally maximal monotone around $(0, 0)$, while T_2 is hypomonotone around $(0, 0)$. Then the graphs of T_1 and T_2 must coincide locally around $(0, 0)$.*

Proof. Let $\bar{T}_1$ be a maximal monotone extension of T_1 , as always exists (cf. [1, Prop. 12.6]). By hypothesis, the graphs of $\bar{T}_1$ and T_1 agree in some neighborhood of $(0, 0)$. The local hypomonotonicity of T_2 means the existence of some $\alpha \geq 0$ such that $T_2 + \alpha I$ is monotone around $(0, 0)$. Because the sum of maximal monotone mappings is maximal monotone when the domain of one meets the interior of the domain of the other, cf. [26, Theorem 1], we have $\bar{T}_1 + \alpha I$ maximal monotone. Hence locally, $T_1 + \alpha I$ is maximal monotone around $(0, 0)$. Since $\text{gph}[T_1 + \alpha I] \subset \text{gph}[T_2 + \alpha I]$, it follows that $T_1 + \alpha I = T_2 + \alpha I$ locally, and the same must therefore be true for T_1 and T_2 themselves. $\square$

A key element in the derivation of Theorem 1.3 will be the fact that the subgradient mapping of a prox-regular function has a “localization” T that is hypomonotone; cf. [2, Theorem 3.2]. A *localization* of ∂f around $(\bar{x}, \bar{v}) \in \text{gph } \partial f$ is a mapping whose graph is obtained by intersecting $\text{gph } \partial f$ with a neighborhood of $(\bar{x}, \bar{v})$.

Throughout the proof, “proto-derivatives” and “second-order epi-derivatives” will be employed. A set-valued mapping $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is *proto-differentiable* at a point x for an element $v \in T(x)$, as defined in [27], if the set-valued mappings

$$\Delta_t T(x|v) : \xi \mapsto [T(x + t\xi) - v]/t \quad \text{for } t > 0$$

graph-converge as $t \downarrow 0$ (i.e., one has set convergence of the graphs). If so, the limit mapping, denoted by $DT(x|v)$, is the *proto-derivative* of T at x for v . On the other hand, a function f is *twice epi-differentiable* at x for a subgradient $v \in \partial f(x)$, as defined in [10], if the second-order difference quotient functions

$$\Delta_t^2 f(x|v) : \xi \mapsto [f(x + t\xi) - f(x) - t\langle v, \xi \rangle] / \frac{1}{2}t^2 \quad \text{for } t > 0$$

epi-converge to a proper function as $t \downarrow 0$ (i.e., one has set convergence of the epigraphs). The epi-limit is then the *second epi-derivative* function $d^2 f(x|v) : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$.

Proof that (a) implies (d). Let $X = \{x \mid |x - \bar{x}| \leq \delta\}$ (where δ is given in Definition 1.1) and $g := (f + \delta_X)^*$; i.e.,

$$g(v) = \max_{x \in X} \{\langle x, v \rangle - f(x)\}.$$

This function g is proper, lsc, convex, and finite. Let

$$G(v) := \operatorname{argmax}_{x \in X} \{\langle x, v \rangle - f(x)\}.$$

Under our assumptions, G is single-valued and Lipschitzian on V , and one can easily verify that $G(v) \in \partial g(v)$. Moreover, the mapping G is monotone on V . Indeed, if $x_i \in G(v_i)$ with $v_i \in V$, then

$$\begin{aligned} \langle x_1 - x_2, v_1 - v_2 \rangle &= \langle x_1, v_1 \rangle - \langle x_2, v_1 \rangle - \langle x_1, v_2 \rangle + \langle x_2, v_2 \rangle \\ &= g(v_1) + f(x_1) - \langle x_2, v_1 \rangle - \langle x_1, v_2 \rangle + g(v_2) + f(x_2) \\ &= [g(v_1) - \langle x_2, v_1 \rangle + f(x_2)] + [g(v_2) - \langle x_1, v_2 \rangle + f(x_1)] \geq 0. \end{aligned}$$

It follows that $\partial g(v) = G(v)$ for all $v \in V$, inasmuch as ∂g is a monotone mapping, whereas G is single-valued continuous and monotone on V , hence maximal monotone relative to V (cf. [1, Ex. 12.7, Ex. 12.48]).

Therefore, g is a convex function that is differentiable on the open neighborhood V of 0, with ∇g Lipschitzian there and satisfying $\nabla g(0) = 0$ (because $G(0) = 0$), as well as $\nabla g(v) \in \operatorname{int} X$ for $v \in V$, and

$$(3.2) \quad \nabla g(v) \in [\partial f]^{-1}(v) \quad \text{for } v \in V.$$

Also, because $\nabla g(v)$ gives the argmax in the conjugate formula for g , we have

$$(3.3) \quad f(x) = \langle v, x \rangle - g(v) \text{ when } v \in V, x = \nabla g(v).$$

Denote by h the function conjugate to g ; thus, $h(x) = \sup_{v \in \mathbb{R}^n} \{ \langle x, v \rangle - g(v) \}$.

The mapping $\partial g : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is maximal monotone, and its inverse $[\partial g]^{-1}$, which is ∂h , is maximal monotone as well; cf. [28]. Note that $0 \in \partial h(0)$ and that $h(0) = 0$. Let T_0 be the localization of ∂h obtained by inverting ∂g relative to V ; thus, T_0^{-1} is ∇g on V but empty-valued outside of V . According to (3.3), the function $(x, v) \mapsto f(x)$ is continuous on the graph of T_0 . Since $\text{gph } T_0 \subset \text{gph } \partial f$ by (3.2), any localization T of ∂f around $(0, 0)$ must include a localization of T_0 , which in turn is locally maximal monotone around $(0, 0)$. The prox-regularity of f allows us to choose the localization T to be hypomonotone; cf. [2, Theorem 3.2]. We conclude from Lemma 3.1 that the graphs of T and T_0 must agree around $(0, 0)$.

Let $\kappa > 0$ be a Lipschitzian constant for ∇g on V . We will now demonstrate that T_0 , or at least some localization of T_0 around $(0, 0)$, is strongly monotone with constant κ^{-1} , so that T must have this property, too. This will conclude the proof of (a) $\Rightarrow$ (d). The tool will be [2, Proposition 5.7] as applied to h . It gives the criterion that at each point $(x, v) \in \text{gph } T_0$ in some neighborhood of $(0, 0)$ where the graph of the proto-derivative of T_0 is an n -dimensional subspace of $\mathbb{R}^{2n}$, the proto-derivative is strongly monotone with constant κ^{-1} .

The points $(x, v) \in \text{gph } T_0$ where the graph of the proto-derivative is an n -dimensional subspace are precisely the points (v, x) where the graph of the proto-derivative of T_0^{-1} is an n -dimensional subspace. We now argue that at such a point v the function g has a symmetric Hessian matrix $\nabla^2 g(v)$. To see this, recall that the function ∇g is Lipschitzian on V . Hence at points $v \in V$ where the proto-derivative of the mapping T_0^{-1} exists, the domain of the proto-derivative is the whole space (because the set-valued mapping whose graph is given by $\limsup_{t \downarrow 0} [\text{gph } \nabla g - (v, \nabla g(v))]/t$ has the whole space as its domain). Therefore, the second-order epi-derivative of the function g at the point $(v, \nabla g(v))$ has the whole space as its domain, which means that it is purely quadratic and that g has a “second-order expansion” at v ; cf. [3, Theorem 3.1]. A convex function (or more generally any lower- C^2 function) with the above properties at the point v is necessarily twice differentiable there and has a symmetric Hessian matrix $\nabla^2 g(v)$; cf. [3, Theorem 3.2].

Based on the above observation, [2, Proposition 5.7] therefore gives the criterion that for each pair (v, x) such that $v \in V$, $x = \nabla g(v)$, and the Hessian $\nabla^2 g(v)$ exists, the inverse of the linear transformation $w \mapsto \nabla^2 g(v)w$ is strongly monotone with constant κ^{-1} . Because κ is a Lipschitzian constant for ∇g , the norm of the Hessian matrix $\nabla^2 g(v)$ is bounded by κ where the Hessian exists. Then

$$(3.4) \quad \langle z, w \rangle \geq \kappa^{-1} |z|^2 \text{ when } z = \nabla^2 g(v)w,$$

as can be seen for instance through a diagonalization based on the symmetry of the Hessian. The inequality (3.4) means that the (possibly multivalued) inverse of the mapping $w \mapsto \nabla^2 g(v)w$ is strongly monotone with constant κ^{-1} , as required.

Proof that (d) implies (c). This implication is obvious.

Proof that (c) implies (b). Let T be the localization of ∂f around $(0, 0)$ that is strongly monotone with constant η in addition to being maximal monotone locally. Let S be a maximal monotone extension of the monotone mapping $T - \eta I$. The mapping $S + \eta I$ is maximal monotone (again by [26, Theorem 1]) and has a localization

around $(0, 0)$ that includes T . The local max monotonicity of T implies that the graphs of T and $S + \eta I$ must actually agree around $(0, 0)$. Therefore,

$$(3.5) \quad \partial^2 f(0|0) = D^*T(0|0) = D^*S(0|0) + \eta I.$$

According to Theorem 2.1 we have $\langle x', v' \rangle \geq 0$ when $v' \in D^*S(0|0)(x')$. It follows then from (3.5) that $\langle x', v' \rangle \geq \eta|x'|^2$ when $v' \in \partial^2 f(0|0)(x')$. This yields (d), as needed.

Proof that (b) implies (a). We first show that a localization of ∂f around $(0, 0)$ is strongly monotone as well as locally max monotone; this shows that $(\partial f)^{-1}$ is locally single-valued and Lipschitzian near 0.

Choose a localization T of ∂f around $(0, 0)$ that is not only hypomonotone but is such that $\text{gph } T$ is also a graphically Lipschitzian manifold of dimension n around $(0, 0)$. The fact that $\text{gph } T$ can be chosen as such a manifold was noted in [2, Theorem 4.7]. Graphically Lipschitzian manifolds were introduced in [29].

Because $\text{gph } T$ is a Lipschitzian manifold we know that $\text{gph } T$ has a linear tangent space at almost every pair $(x, v) \in \text{gph } T$ in some neighborhood of $(\bar{x}, \bar{v})$; cf. [29]. Moreover, at such points (x, v) , the graph of $DT(x|v)$ is an n -dimensional subspace of $\mathbb{R}^n \times \mathbb{R}^n$; again cf. [29]. Consequently, we define $R : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ to be the set-valued mapping whose graph is given by the limsup of $\text{gph } DT(x|v)$ over the set of points $(x, v) \in \text{gph } T$ converging to $(0, 0)$ and such that $\text{gph } DT(x|v)$ is an n -dimensional subspace of $\mathbb{R}^n \times \mathbb{R}^n$.

Claim. $\text{gph } R \subset \text{gph } \partial^2 f(0|0)$.

Justification. Recall that a function $h : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ is a *generalized quadratic function* if it is expressible in the form

$$h(x) = \begin{cases} \frac{1}{2}\langle x, Qx \rangle & \text{if } x \in N, \\ \infty & \text{if } x \notin N, \end{cases}$$

where N is a linear subspace of $\mathbb{R}^n$ and Q is a symmetric $n \times n$ matrix. According to [3, Theorem 3.9], the second-order epi-derivative is generalized quadratic at each point $(x, v) \in \text{gph } T$ near $(0, 0)$, where $\text{gph } DT(x|v)$ is an n -dimensional subspace (simply apply [3, Theorem 3.9] to the function $f(x) = f(x) + (r/2)|x|^2$ (r given by (3.1)); this function meets all the requirements of that theorem). We know that the proto-derivative of the mapping T at (x, v) gives the subgradient mapping of the second-order epi-derivative; cf. [2, Theorem 6.1]. This implies that

$$(3.6) \quad DT(x|v)(w) = \begin{cases} A_{x,v}(w) + N_{x,v}^T & \text{if } w \in N_{x,v}, \\ \emptyset & \text{otherwise} \end{cases}$$

for some symmetric matrix $A_{x,v}$ and a subspace $N_{x,v}$.

We will now verify that $\text{gph } DT(x|v) \subset \text{gph } \partial^2 f(x|v)$. Take $\bar{w} \in N_{x,v}$ and $\bar{u} \in N_{x,v}^T$. For $w \in N_{x,v}$ and $u \in N_{x,v}^T$ we have

$$\begin{aligned} & \langle A_{x,v}(\bar{w}) + \bar{u}, -\bar{w} \rangle, \langle w, A_{x,v}(w) + u \rangle \\ &= \langle A_{x,v}(\bar{w}), w \rangle + \langle \bar{u}, w \rangle - \langle \bar{w}, A_{x,v}(w) \rangle - \langle \bar{w}, u \rangle = 0 \end{aligned}$$

(remember here that $A_{x,v}$ is symmetric). This confirms that the vector $(A_{x,v}(\bar{w}) + \bar{u}, -\bar{w})$ belongs to the normal cone to $\text{gph } T$ at (x, v) , or in other words that $(\bar{w}, A_{x,v}(\bar{w}) + \bar{u})$ lies in $\text{gph } \partial^2 f(x|v)$. (The normal cone includes the polar of the tangent cone, which in this case is the graph of the proto-derivative $DT(x|v)$; cf. [1].)

We then have the desired result by the definition of the mapping R and the fact that $\limsup$ of $\text{gph } \partial^2 f(x|v)$ over the set of points $(x, v) \in \text{gph } T$ converging to $(0, 0)$ is a subset of $\text{gph } \partial^2 f(0|0)$; cf. [23] and [1]. This finishes the justification of the claim.

Continuing with the proof of (b) $\Rightarrow$ (a) in Theorem 1.3, we note that, by our assumptions, one has $\langle z, w \rangle > 0$ for all $z \in R(w)$ with $w \neq 0$ (the latter being true for all $z \in \partial^2 f(0|0)(w)$ with $w \neq 0$). We will show that this condition implies the existence of $\epsilon > 0$ such that, for all $\delta \in (0, \epsilon)$ and $(x, v) \in \text{gph } T$ close enough to $(0, 0)$ with $\text{gph } DT(x|v)$ an n -dimensional subspace, one has

$$(3.7) \quad \langle z, w \rangle \geq \delta |w|^2 \text{ when } z \in DT(x|v)(w) \text{ with } w \neq 0.$$

Recall that for such (x, v) , $DT(x|v)$ has a representation in the form of (3.6). Actually, in that representation we may choose the symmetric matrix $A_{x,v}$ with the property that $A_{x,v} : N_{x,v} \rightarrow N_{x,v}$; cf. [29, Proof of Proposition 4.1]. (In the cited proof, it is shown that if the subgradient mapping of a closed proper convex function is a subspace, then the subgradient mapping has the desired representation. Our case follows from the convex situation because the second-order epi-derivative of a prox-regular function has the property that $d^2 f(x|v) + \rho |\cdot|^2$ is convex for some $\rho > 0$; cf. [1, Thm. 13.49] or [2, Thm. 6.5].) Now suppose there exist $(x_\epsilon, v_\epsilon) \in \text{gph } T$ converging to $(0, 0)$ as $\epsilon \rightarrow 0$ with $DT(x|v)$ an n -dimensional subspace and $w_\epsilon \in N_{x_\epsilon, v_\epsilon}$ with $|w_\epsilon| = 1$ and $|A_{x_\epsilon, v_\epsilon}(w_\epsilon)| < \epsilon$. When this is true, then by taking a cluster point we can find w of norm 1 with $0 \in R(w)$ a contradiction. Therefore, there exists $\epsilon > 0$ such that for each $(x, v) \in \text{gph } T$ close to $(0, 0)$ with $DT(x|v)$ an n -dimensional subspace, the subspace $N_{x,v}$ contains only the origin or the minimum eigenvalue of $A_{x,v}$ is greater than ϵ . From this we have (3.7).

Next, from (3.7) we conclude that, for each $\delta \in (0, \epsilon)$, $DT(x|v)$ is strongly monotone with constant δ . From [2, Proposition 5.7] we see that T is strongly monotone with constant δ .

To complete the proof of the theorem, we need only establish the existence of a neighborhood X of 0 such that the mapping $v \mapsto \text{argmin}_{x \in X} \{f(x) - \langle v, x \rangle\}$ coincides around 0 with the mapping $v \mapsto (\partial f)^{-1}(v) \cap X$. This is accomplished through [2, Theorem 5.5] and the fact that T is strongly monotone. Using [2, Theorem 5.5] we conclude that for some $k > 0$ and $\lambda > 0$, $e_\lambda - k|\cdot|^2$ is convex on a neighborhood of 0. Here e_λ is the Moreau-envelope of parameter $\lambda > 0$; i.e.,

$$e_\lambda(x) = \min_{x'} \left\{ f(x') + \frac{1}{2\lambda} |x' - x|^2 \right\}.$$

This shows that $e_\lambda(x) \geq k|x|^2$, which implies that $f(x) \geq k|x|^2$ on a neighborhood of 0 (because $f(x) \geq e_\lambda(x)$). From the previous fact we can find neighborhoods X and V of 0 such that, for $v \in V$, the set $\text{argmin}_{x \in X} \{f(x) - \langle v, x \rangle\}$ is nonempty and is in the interior of X . For such v , $\text{argmin}_{x \in X} \{f(x) - \langle v, x \rangle\} \subset (\partial f)^{-1}(v)$. The inclusion must then be an equation, because the potentially bigger set is just a singleton. $\square$

Remark 3.2. According to [30, Theorem 1.2], the mapping M is hypomonotone and has the Aubin property near $(0, \bar{x}) \in \text{gph } M$ when $\bar{x}$ gives a tilt-stable local minimum. The Aubin property refers to a Lipschitzian-like property which is formulated for mappings that might be set-valued.

4. Application to constrained minimization. Let $f_0 : \mathbb{R}^n \rightarrow \mathbb{R}$ be a C^2 function and let C be a nonempty, closed subset of $\mathbb{R}^n$. As already noted, the minimization of f_0 over C corresponds to the minimization of $f = f_0 + \delta_C$ over $\mathbb{R}^n$, where

δ_C is the indicator of C . In this case, $\bar{x}$ gives a tilt-stable minimum of f if and only if there exists $\delta > 0$ such that the mapping

$$(4.1) \quad M : v \mapsto \operatorname{argmin}\{f_0(x) - f_0(\bar{x}) - \langle v, x - \bar{x} \rangle \mid x \in C, |x - \bar{x}| \leq \delta\}$$

is single-valued and Lipschitzian on some neighborhood of $v = 0$. It is appropriate to speak then of $\bar{x}$ giving a *tilt-stable local minimum of f_0 relative to C* . What does Theorem 1.3 say about that property?

In this setting, we have $\partial f(\bar{x}) = \nabla f_0(\bar{x}) + \partial \delta_C(\bar{x})$. On the level of the first-order condition for optimality, therefore, we have

$$(4.2) \quad 0 \in \partial f(\bar{x}) \iff -\nabla f_0(\bar{x}) \in \partial \delta_C(\bar{x}) \iff -\nabla f_0(\bar{x}) \in N_C(\bar{x}).$$

To proceed to the level of the second-order condition, we can appeal to the notion of C being prox-regular at $\bar{x}$ for a vector $\bar{v}$, which by definition is equivalent to the function δ_C being prox-regular at $\bar{x}$ for $\bar{v}$; see [2], where this class of sets is discussed in some detail.

PROPOSITION 4.1. *If $f = f_0 + \delta_C$ with f_0 of class $\mathcal{C}^2$ and C closed, one has for any $\bar{x} \in C$ and $w \in \mathbb{R}^n$ that*

$$(4.3) \quad \partial^2 f(\bar{x}|0)(w) = \nabla^2 f_0(\bar{x})w + \partial^2 \delta_C(\bar{x}|\bar{v})(w) \quad \text{for } \bar{v} = -\nabla f_0(\bar{x}).$$

Furthermore, f is prox-regular and subdifferentially continuous at $\bar{x}$ if and only if C is prox-regular at $\bar{x}$ for $\bar{v}$.

Proof. By a calculus result of Mordukhovich [23], a mapping of the form $T(x) = F(x) + S(x)$ with F single-valued and of class $\mathcal{C}^1$ has $D^*T(x|v) = \nabla F(x) + D^*S(x|v - F(x))$. Here we apply this to $F = \nabla f_0$ and $S = \partial \delta_C$. Trivially, the function δ_C is subdifferentially continuous. But from [1, Prop. 13.35], the prox-regularity of a function is preserved when a $\mathcal{C}^2$ function is added; the same holds for subdifferential continuity as well. $\square$

THEOREM 4.2. *For a $\mathcal{C}^2$ function f_0 on $\mathbb{R}^n$ and a closed set $C \subset \mathbb{R}^n$, let $\bar{x}$ be a point of C satisfying the first-order condition $-\nabla f_0(\bar{x}) \in \partial \delta_C(\bar{x})$ and suppose that C is prox-regular at $\bar{x}$ (i.e., that δ_C is prox-regular there). Then $\bar{x}$ gives a tilt-stable local minimum of f_0 relative to C if and only if, in terms of $\bar{v} = -\nabla f_0(\bar{x})$, one has*

$$(4.4) \quad \langle w, \nabla^2 f_0(\bar{x})w \rangle > -\langle z, w \rangle \quad \text{whenever } z \in \partial^2 \delta_C(\bar{x}|\bar{v})(w), w \neq 0.$$

Proof. This is evident from the equivalence of (a) and (b) of Theorem 1.3 in the context of Proposition 4.1. $\square$

When the set C is convex, the coderivative result in Theorem 2.1 can be brought into this picture.

COROLLARY 4.3. *For a $\mathcal{C}^2$ function f_0 and a closed, convex set C , let $\bar{x}$ be a point satisfying the first-order condition $-\nabla f_0(\bar{x}) \in \partial \delta_C(\bar{x})$. Then, with $\bar{v} = -\nabla f_0(\bar{x})$, a sufficient condition for $\bar{x}$ to give a tilt-stable local minimum of f_0 relative to C is that*

$$(4.5) \quad \langle w, \nabla^2 f_0(\bar{x})w \rangle > 0 \quad \text{whenever } w \in \operatorname{dom} \partial^2 \delta_C(\bar{x}|\bar{v}), w \neq 0.$$

Proof. The convexity of C implies that the normal cone mapping $N_C = \partial \delta_C$ is maximal monotone; cf. [28]. Then $\langle z, w \rangle \geq 0$ by Theorem 2.1 whenever $z \in \partial^2 \delta_C(\bar{x}|\bar{v})(w)$. Condition (4.5) is enough in that case to ensure that condition (4.4) holds. $\square$

When C is not just convex but also polyhedral, an exact formula for the generalized Hessian $\partial^2 \delta_C(\bar{x} | \bar{v})$ is available from [31]. In order to state this formula, we need to deal with the *critical cone* for C at $\bar{x}$ for a vector $\bar{v} \in \partial \delta_C(\bar{x}) = N_C(\bar{x})$; this is the polyhedral convex cone

$$K(\bar{x}, \bar{v}) = \{ w \in T_C(\bar{x}) \mid w \perp \bar{v} \},$$

where $T_C(\bar{x})$ denotes the tangent cone to C at $\bar{x}$. Recall that a *closed face* F of a polyhedral convex cone K is a polyhedral convex cone of the form

$$F = \{ x \in K \mid x \perp v \} \quad \text{for some } v \in K^*,$$

where K^* denotes the polar of K .

PROPOSITION 4.4. *For a polyhedral convex set C , the generalized Hessian of the indicator of C at $\bar{x}$ for any $\bar{v} \in \partial \delta_C(\bar{x})$ is given, with respect to the critical cone $K(\bar{x}, \bar{v})$, by*

$$(4.6) \quad z \in \partial^2 \delta_C(\bar{x} | \bar{v})(w) \iff \begin{cases} \text{there exist closed faces } F_1 \text{ and } F_2 \text{ of } K(\bar{x}, \bar{v}) \\ \text{with } F_2 \subset F_1, w \in F_2 - F_1, z \in (F_1 - F_2)^*. \end{cases}$$

Proof. In [31, Proof of Theorem 2], the normal cone to $\text{gph } N_C = \text{gph } \partial \delta_C$ at the point $(\bar{x}, \bar{v})$ is shown to be the union of products sets $K^* \times K$ over all the cones $K = F_1 - F_2$, where F_1 and F_2 are described as above. $\square$

THEOREM 4.5. *For a $\mathcal{C}^2$ function f_0 and a polyhedral convex set C , let $\bar{x}$ be a point satisfying the first-order condition $-\nabla f_0(\bar{x}) \in \partial \delta_C(\bar{x})$. Then, with respect to $\bar{v} = -\nabla f_0(\bar{x})$ and the critical cone $K(\bar{x}, \bar{v})$, a necessary and sufficient condition for $\bar{x}$ to give a tilt-stable local minimum of f_0 relative to C is that*

$$(4.7) \quad \langle w, \nabla^2 f_0(\bar{x}) w \rangle > 0 \text{ for all } w \neq 0 \text{ in } K(\bar{x}, \bar{v}) - K(\bar{x}, \bar{v}).$$

Proof. On the basis of Theorem 4.2 and Proposition 4.4, we know that $\bar{x}$ gives a tilt-stable local minimum if and only if $\langle w, \nabla^2 f_0(\bar{x}) w \rangle > -\langle z, w \rangle$ whenever $w \neq 0$ and there exist closed faces F_1 and F_2 of $K(\bar{x}, \bar{v})$ with $F_2 \subset F_1$, such that $w \in F_2 - F_1$ and $z \in (F_1 - F_2)^*$. In those circumstances one can always take $z = 0$, so the condition comes down to having $\langle w, \nabla^2 f_0(\bar{x}) w \rangle > 0$ for all $w \neq 0$ in the union of all cones $F_2 - F_1$ generated by faces of $K(\bar{x}, \bar{v})$, as described. But all such faces lie in $K(\bar{x}, \bar{v})$, which in particular is a closed face of itself. Therefore, the union is simply $K(\bar{x}, \bar{v}) - K(\bar{x}, \bar{v})$. $\square$

Note that $K(\bar{x}, \bar{v}) - K(\bar{x}, \bar{v})$ is a subspace of $\mathbb{R}^n$, the smallest one that includes $K(\bar{x}, \bar{v})$. When the critical cone $K(\bar{x}, \bar{v})$ is itself a subspace, condition (4.7) is the same as

$$\langle w, \nabla^2 f_0(\bar{x}) w \rangle > 0 \text{ for all } w \neq 0 \text{ in } K(\bar{x}, \bar{v}),$$

which is the usual second-order sufficient condition for optimality in this framework of polyhedral convexity.

Elaborations of Theorem 4.2 for other, nonpolyhedral classes of sets C must await further advances in the calculus of generalized Hessians.

Acknowledgments. The first author wishes to thank D. Azé and J. N. Corvellec for their generous hospitality. The authors are indebted to A. Levy for his careful reading of an earlier version of this paper.

REFERENCES

- [1] R. T. ROCKAFELLAR AND R. J.-B. WETS, *Variational Analysis*, Springer-Verlag, New York, 1997.
- [2] R. A. POLIQUIN AND R. T. ROCKAFELLAR, *Prox-regular functions in variational analysis*, Trans. Amer. Math. Soc., 348 (1996), pp. 1805–1838.
- [3] R. A. POLIQUIN AND R. T. ROCKAFELLAR, *Generalized Hessian properties of regularized nonsmooth functions*, SIAM J. Optim., 6 (1996), pp. 1121–1137.
- [4] R. A. POLIQUIN AND R. T. ROCKAFELLAR, *Second-order nonsmooth analysis in nonlinear programming*, in Recent Advances in Nonsmooth Optimization, D. Du, L. Qi, and R. Womersley, eds., World Scientific Publishers, River Edge, NJ, 1995, pp. 322–350.
- [5] R. A. POLIQUIN AND R. T. ROCKAFELLAR, *Amenable functions in optimization*, in Nonsmooth Optimization Methods and Applications, F. Giannessi, ed., Gordon and Breach, Philadelphia, PA, 1992, pp. 338–353.
- [6] R. A. POLIQUIN AND R. T. ROCKAFELLAR, *A calculus of epi-derivatives applicable to optimization*, Canadian J. Math., 45 (1993), pp. 879–896.
- [7] R. A. POLIQUIN AND R. T. ROCKAFELLAR, *Proto-derivatives formulas for basic subgradient mappings in mathematical programming*, Set-Valued Anal., 2 (1994), pp. 275–290.
- [8] A. LEVY AND R. T. ROCKAFELLAR, *Variational conditions and the proto-differentiation of partial subgradient mappings*, Nonlinear Anal., 26 (1996), pp. 1951–1964.
- [9] J. V. BURKE, *Second order necessary and sufficient conditions for convex composite NDO*, Math. Prog., 38 (1987), pp. 287–302.
- [10] R. T. ROCKAFELLAR, *First- and second-order epi-differentiability in nonlinear programming*, Trans. Amer. Math. Soc., 307 (1988), pp. 75–107.
- [11] R. T. ROCKAFELLAR, *Second-order optimality conditions in nonlinear programming obtained by way of epi-derivatives*, Math. Oper. Res., 14 (1989), pp. 462–484.
- [12] R. A. POLIQUIN, *Proto-differentiation of subgradient set-valued mappings*, Canadian J. Math., 42 (1990), pp. 520–532.
- [13] R. COMINETTI, *On Pseudo-differentiability*, Trans. Amer. Math. Soc., 324 (1991), pp. 843–865.
- [14] A. D. IOFFE, *Variational analysis of a composite function: A formula for the lower second order epi-derivative*, J. Math. Anal. Appl., 160 (1991), pp. 379–405.
- [15] R. A. POLIQUIN, *Integration of subdifferentials of nonconvex functions*, Nonlinear Anal. Th. Meth. Appl., 17 (1991), pp. 385–398.
- [16] J. V. BURKE AND R. A. POLIQUIN, *Optimality conditions for non-finite convex composite functions*, Math. Prog., 57 (1992), pp. 103–120.
- [17] R. A. POLIQUIN, *An extension of Attouch's Theorem and its application to second-order epi-differentiation of convexly composite functions*, Trans. Amer. Math. Soc., 332 (1992), pp. 861–874.
- [18] R. A. POLIQUIN, J. VANDERWERFF, AND V. ZIZLER, *Convex composite representation of lower semicontinuous functions and renormings*, C. R. Acad. Sci. Paris Sér. I, t. 317 (1993), pp. 545–549.
- [19] A. LEVY, R. A. POLIQUIN, AND L. THIBAUT, *Partial extension of Attouch's theorem with applications to proto-derivatives of subgradient mappings*, Trans. Amer. Math. Soc., 347 (1995), pp. 1269–1294.
- [20] B. S. MORDUKHOVICH, *Sensitivity analysis in nonsmooth optimization*, in Theoretical Aspects of Industrial Design, D. A. Field and V. Komkov, eds., SIAM Volumes in Applied Mathematics, 58 (1992), pp. 32–46.
- [21] B. S. MORDUKHOVICH, *Lipschitzian stability of constraint systems and generalized equations*, Nonlinear Anal., 22 (1994), pp. 173–206.
- [22] B. S. MORDUKHOVICH, *Complete characterization of openness, metric regularity, and Lipschitzian properties of multifunctions*, Trans. Amer. Math. Soc., 340 (1993), pp. 1–36.
- [23] B. S. MORDUKHOVICH, *Generalized differential calculus for nonsmooth and set-valued mappings*, J. Math. Anal. Appl., 183 (1994), pp. 250–288.
- [24] A. D. IOFFE, *Approximate subdifferentials and applications, I: The finite dimensional theory*, Trans. Amer. Math. Soc., 281 (1984), pp. 389–416.
- [25] R. T. ROCKAFELLAR, *Favorable classes of Lipschitz continuous functions in subgradient optimization*, in Progress in Nondifferentiable Optimization, E. Nurminski, ed., IIASA Collaborative Proceedings Series, International Institute of Applied Systems Analysis, Laxenburg, Austria, 1982, pp. 125–144.
- [26] R. T. ROCKAFELLAR, *On the maximality of sums of nonlinear monotone operators*, Trans. Amer. Math. Soc., 149 (1970), pp. 75–88.

- [27] R. T. ROCKAFELLAR, *Proto-differentiability of set-valued mappings and its applications in optimization*, in *Analyse Non Linéaire*, H. Attouch et al., eds., Gauthier-Villars, Paris, 1989, pp. 449–482.
- [28] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
- [29] R. T. ROCKAFELLAR, *Maximal monotone relations and the second derivatives of nonsmooth functions*, *Ann. Inst. H. Poincaré Anal. Non Linéaire*, 2 (1985), pp. 167–184.
- [30] A. B. LEVY AND R. A. POLIQUIN, *Characterizations of the local single-valuedness of multifunctions*, *Set-Valued Anal.*, to appear.
- [31] A. DONTCHEV AND R. T. ROCKAFELLAR, *Characterizations of strong regularity for variational inequalities over polyhedral convex sets*, *SIAM J. Optim.*, 6 (1996), pp. 1087–1105.

A CHAIN RULE FOR ESSENTIALLY SMOOTH LIPSCHITZ FUNCTIONS*

JONATHAN M. BORWEIN[†] AND WARREN B. MOORS[‡]

Abstract. In this paper we introduce a new class of real-valued locally Lipschitz functions (that are similar in nature and definition to Valadier's *saine* functions), which we call *arcwise essentially smooth*, and we show that if $g : R^m \rightarrow R$ is arcwise essentially smooth on R^m and each function $f_j : R^n \rightarrow R$, $1 \leq j \leq m$, is strictly differentiable almost everywhere in R^n , then $g \circ f$ is strictly differentiable almost everywhere in R^n , where $f \equiv (f_1, f_2, \dots, f_m)$. We also show that all the semismooth and all the pseudoregular functions are arcwise essentially smooth. Thus, we provide a large and robust lattice algebra of Lipschitz functions whose generalized derivatives are well behaved.

Key words. Lipschitz functions, chain rule, Haar-null sets, differentiability, essentially smooth

AMS subject classifications. Primary, 49J520, 46N10; Secondary, 58C20

PII. S1052623496297838

1. Introduction. In this paper we show that those Lipschitz functions which are strictly differentiable almost everywhere (that is, *essentially smooth*) possess extremely strong closure properties. Essentially smooth functions were studied in great detail in [3] and [4]. In particular, it was shown therein that all such functions possess very well-behaved Clarke generalized gradients.

We begin by recalling some preliminary definitions regarding the Clarke subdifferential mapping [6]. A real-valued function f defined on a nonempty open subset U of a Banach space X is *locally Lipschitz* on U if for each $x_0 \in U$ there exists a $K > 0$ and a $\delta > 0$ such that

$$|f(x) - f(y)| \leq K\|x - y\| \quad \text{for all } x, y \in B(x_0, \delta).$$

For functions in this class, it is often instructive to consider the following three directional derivatives:

(i) The *upper Dini derivative* at $x \in U$, in the direction y , is given by

$$f^+(x; y) \equiv \limsup_{\lambda \rightarrow 0^+} \frac{f(x + \lambda y) - f(x)}{\lambda}.$$

(ii) The *lower Dini derivative* at $x \in U$, in the direction y , is given by

$$f^-(x; y) \equiv \liminf_{\lambda \rightarrow 0^+} \frac{f(x + \lambda y) - f(x)}{\lambda}.$$

(iii) The *Clarke generalized directional derivative* at $x \in U$, in the direction y , is given by

$$f^0(x; y) \equiv \limsup_{\substack{z \rightarrow x \\ \lambda \rightarrow 0^+}} \frac{f(z + \lambda y) - f(z)}{\lambda}.$$

*Received by the editors January 26, 1996; accepted for publication (in revised form) February 21, 1997.

<http://www.siam.org/journals/siopt/8-2/29783.html>

[†]CECM, Department of Mathematics and Statistics, Simon Fraser University, Burnaby, BC V5A 1S6, Canada (jborwein@cecm.sfu.ca). The research of this author was supported by NSERC and the Shrum Endowment at Simon Fraser University.

[‡]Department of Mathematics, The University of Auckland, Private Bag 92019, Auckland, New Zealand (moors@math.auckland.ac.nz). The research of this author was supported by a New Zealand Science and Technology Post-Doctoral Fellowship.

It is immediate from these three definitions that for each $x \in U$ and each $y \in X$,

$$f^-(x; y) \leq f^+(x; y) \leq f^0(x; y).$$

Let us now examine some notions of differentiability that are associated with locally Lipschitz functions. We say that a real-valued locally Lipschitz function f is *differentiable at x , in the direction y* , if

$$f'(x; y) \equiv \lim_{\lambda \rightarrow 0} \frac{f(x + \lambda y) - f(x)}{\lambda} \text{ exists.}$$

We say that f is *Gâteaux differentiable at x* if

$$\nabla f(x)(y) \equiv \lim_{\lambda \rightarrow 0} \frac{f(x + \lambda y) - f(x)}{\lambda} \text{ exists}$$

for each $y \in X$ and $\nabla f(x)$ is a continuous linear functional on X . In this paper we will also be interested in two slightly stronger notions of differentiability. A locally Lipschitz function f is said to be *strictly differentiable at x , in the direction y* , if

$$\lim_{\substack{z \rightarrow x \\ \lambda \rightarrow 0^+}} \frac{f(z + \lambda y) - f(z)}{\lambda} \text{ exists,}$$

and we say that f is *strictly differentiable at x* , if f is strictly differentiable at x , in every direction $y \in X$. Let us recall that a function f is strictly differentiable at x , in the direction y , if and only if

$$f^0(x; y) = f'(x; y) = -f^0(x; -y).$$

In addition to the various notions of differentiability we will also need the notion of a “null” set. Let $(X, \|\cdot\|)$ be a Banach space. We will call a Borel subset $N \subseteq X$ a *Haar-null* set if there exists a (not necessarily unique) Radon probability measure p on X such that $p(x + N) = 0$ for each $x \in X$. (In this case, we call p a *test measure* for N .)

More generally, we say that a subset $N \subseteq X$ is a *Haar-null* set if it is contained in a Borel Haar-null set (see [4] for a slightly more general definition of Haar-null). Further, we will say that a property P holds *almost everywhere* if $\{x \in X : P(x) \text{ is not true}\}$ is a Haar-null set. The Haar-null sets are closed under translation and countable unions. Moreover, no nonempty open subset is Haar-null [4]. In finite dimensions, the Haar-null sets coincide with the Lebesgue-null sets.

A further important class of sets that we will need to consider is the following. Let U be a nonempty open subset of a Banach space X . Then a Borel subset S of U is *1 - D almost everywhere in U , in the direction y* , if for each $x \in U$

$$\mu(\{t \in R : x + ty \in U \text{ and } x + ty \notin S\}) = 0.$$

(Here, and later, μ will denote the Lebesgue measure on R .)

For us, the most important example of a 1 - D almost everywhere set is the following.

PROPOSITION 1.1 (see [8, Remark 2.4]). *Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a Banach space X . Then for each $y \in X$, $D_y \equiv \{x \in U : f'(x; y) \text{ exists}\}$ is 1 - D almost everywhere in U , in the direction y .*

Clearly, if S is $1 - D$ almost everywhere in U , for some direction y , then $U \setminus S$ is a Haar-null set (see Proposition 2.2 in [4] if details are desired).

In [3] the present authors said that a real-valued locally Lipschitz function f defined on a nonempty open subset U of a Banach space X is *essentially smooth* on U if for each $y \in X$

$$\{x \in U : f^0(x; y) \neq -f^0(x; -y)\} \text{ is a Haar-null subset of } X$$

and they denoted by $\mathcal{S}_e(U)$ the family of all real-valued essentially smooth locally Lipschitz functions defined on U .

On a few occasions we will need to consider vector-valued functions. Let U be a nonempty open subset of a Banach space X and let V be a nonempty open subset of R^m . If $f : U \rightarrow V$ is defined by

$$f(x) \equiv (f_1(x), f_2(x), \dots, f_m(x)) \quad \text{with } f_j : U \rightarrow R,$$

then we say that the vector-valued function f is *essentially smooth* on U if $f_j \in \mathcal{S}_e(U)$ for each $1 \leq j \leq m$, and in this case we write $f \in \mathcal{S}_e(U; V)$. In addition, we will say that f is *strictly differentiable* at $x \in U$, in the direction y , if

$$f_j^0(x; y) = -f_j^0(x; -y) \quad \text{for each } 1 \leq j \leq m.$$

Let us now return to the purpose of this paper which is, as mentioned earlier, to show that the family of functions $\mathcal{S}_e(U)$ possesses very strong closure properties. A first but naive guess might be that if $f_1, f_2, \dots, f_m \in \mathcal{S}_e(U)$ and $g \in \mathcal{S}_e(R^m)$, then $g \circ f \in \mathcal{S}_e(U)$, where $f \equiv (f_1, f_2, \dots, f_m)$. However, the following example shows that in general this is not true (when $m \geq 2$).

Example 1.1. Let X be a Banach space, and let C be a Cantor subset of R with positive Lebesgue measure. Define the functions $f_1, \dots, f_m$ on X by $f_m \in X^* \setminus \{0\}$ and $f_j \equiv 0$ for each $1 \leq j < m$. Further, we define $d_C : R \rightarrow R$ by $d_C(t) \equiv \text{dist}(t, C)$ and $g : R^m \rightarrow R$ by $g(x_1, x_2, \dots, x_m) \equiv \text{dist}((x_1, x_2, \dots, x_m), \{0\} \times \{0\} \times \dots \times C)$ (here the distance is with respect to the Euclidean norm on R^m). Clearly, each f_j is strictly differentiable on X . Moreover, by Theorem 6.2 in [3] we have that $g \in \mathcal{S}_e(R^m)$. We claim that $g \circ f \notin \mathcal{S}_e(X)$, where $f \equiv (f_1, f_2, \dots, f_m)$. To see this, observe that $g \circ f(x) = d_C(f_m(x))$. Now, it is standard that d_C is not strictly differentiable at any point of C . Hence, it follows that $g \circ f$ is not strictly differentiable at any point of $f_m^{-1}(C)$, which is not a Haar-null set (see Theorem 2.3 in [4]). It now follows that $g \circ f \notin \mathcal{S}_e(X)$.

Despite this example, $\mathcal{S}_e(U)$ does possess very strong closure properties.

2. A chain rule. In this section of the paper, we show that there is a large subclass of the essentially smooth functions, which is closed under composition. This class of functions is closely related to Valadier's *saine* functions [9]. Let U be a nonempty open subset of R^m . We say that a real-valued locally Lipschitz function f defined on U is *arcwise essentially smooth* on U if for each locally Lipschitz function $x \in \mathcal{S}_e((0, 1); U)$,

$$\mu(\{t \in (0, 1) : f^0(x(t); x'(t)) \neq -f^0(x(t); -x'(t))\}) = 0;$$

$$\text{here, } x'(t) \equiv (x'_1(t), x'_2(t), \dots, x'_m(t)).$$

(Valadier requires this to hold for *all* absolutely continuous arcs.) We shall denote by $\mathcal{A}_e(U)$ the family of all arcwise essentially smooth functions defined on U .

Remark 2.1. It is easily seen that the definition of arcwise essential smoothness is unaffected by replacing the open set $(0, 1)$ (given in the definition) by any other nonempty open subset of R .

PROPOSITION 2.1. *For each nonempty open subset U of R^m $\mathcal{A}_e(U) \subseteq \mathcal{S}_e(U)$.*

Proof. Consider $f \in \mathcal{A}_e(U)$. For each fixed $y \in S(R^m)$ (the unit sphere in R^m) we will show that the G_δ set, $S_y \equiv \{x \in U : f^0(x; y) = -f^0(x; -y)\}$ is $1 - D$ almost everywhere in U , in the direction y . To this end, consider $x_0 \in U$. Let $U_0 \equiv \{t \in R : x_0 + ty \in U\}$. Clearly U_0 is nonempty and open. Now, consider the mapping $x : U_0 \rightarrow U$ defined by $x(t) \equiv x_0 + ty$. It is obvious that $x \in \mathcal{S}_e(U_0; U)$. We may now perform a little set arithmetic.

$$\begin{aligned} & \{t \in R : x_0 + ty \in U \text{ and } x_0 + ty \notin S_y\} \\ &= \{t \in U_0 : f^0(x(t); x'(t)) \neq -f^0(x(t); -x'(t))\} \end{aligned}$$

and so, $\mu(\{t \in R : x_0 + ty \in U \text{ and } x_0 + ty \notin S_y\}) = 0$, which shows that S_y is $1 - D$ almost everywhere in U , in the direction y . Hence, $U \setminus S_y$ is a Haar-null (Lebesgue-null) set. $\square$

PROPOSITION 2.2 ([3, Theorem 4.4]). *If U is a nonempty open subset of R , then $\mathcal{A}_e(U) = \mathcal{S}_e(U)$.*

Our next task is to show that the class of functions $\mathcal{A}_e(U)$ is reasonably large, above and beyond the obvious observation that $\mathcal{A}_e(U)$ contains all the C^1 functions defined on U .

LEMMA 2.1. *Let U be a nonempty open subset of R^m , and let f be a real-valued locally Lipschitz function defined on U . Then $f \in \mathcal{A}_e(U)$ if and only if for each $x \in \mathcal{S}_e((0, 1); U)$*

$$\mu(\{t \in (0, 1) : f^0(x(t); x'(t)) = f'(x(t); x'(t))\}) = 1.$$

Proof. Let $x \in \mathcal{S}_e((0, 1); U)$. We need to show that

$$\mu(\{t \in (0, 1) : f^0(x(t); x'(t)) = -f^0(x(t); -x'(t))\}) = 1.$$

Consider the mapping $y : (0, 1) \rightarrow U$, defined by $y(t) \equiv x(1 - t)$. Note that almost everywhere $y'(t) = -x'(1 - t)$.

Let $E_1 \equiv \{t \in (0, 1) : f^0(x(t); x'(t)) = f'(x(t); x'(t))\}$ and $E_2 \equiv \{t \in (0, 1) : -f^0(x(t); -x'(t)) = f'(x(t); x'(t))\}$. Clearly, $f^0(x(t); x'(t)) = -f^0(x(t); -x'(t))$ on $E_1 \cap E_2$, so it remains to show that $\mu(E_2) = 1$.

Now, by the hypothesis, $E_3 \equiv \{t \in (0, 1) : f'(y(t); y'(t)) = f^0(y(t); y'(t))\}$ has measure one. On the other hand, $E_2 = h(E_3)$, where $h(t) \equiv 1 - t$. Therefore, $\mu(E_2) = 1$. The converse is obvious. $\square$

The next few lemmas are quite standard, but we include them for the sake of completeness.

LEMMA 2.2. *Let U be a nonempty open subset of a Banach space X and let $f \equiv (f_1, f_2, \dots, f_m)$ be a locally Lipschitz mapping from U into R^m . Furthermore, let g be a real-valued locally Lipschitz function defined on a nonempty open subset V of R^m which contains $f(U)$.*

If f is differentiable at some point x_0 , in the direction y , and either

(i) $(g \circ f)'(x_0; y)$ exists or (ii) $g'(f(x_0); f'(x_0; y))$ exists, then

$$(g \circ f)'(x_0; y) = g'(f(x_0); f'(x_0; y)).$$

Remark 2.2. From Lemma 2.2 and Proposition 1.1 we see that for each fixed $y \in X$, $\{x \in U : (g \circ f)'(x; y) = g'(f(x); f'(x; y))\}$ is $1 - D$ almost everywhere in U , in the direction y .

LEMMA 2.3 ([4, Theorem 3.1]). *Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a Banach space X . If $U \setminus S$ is a Haar-null set, then for each $x \in U$*

$$f^0(x; y) = \limsup_{\substack{z \rightarrow x \\ z \in S}} f^+(z; y) \text{ and } -f^0(x; -y) = \liminf_{\substack{z \rightarrow x \\ z \in S}} f^+(z; y).$$

Remark 2.3. From Lemma 2.3 we see that $f^0(x; y) = -f^0(x; -y)$ if and only if

$$\lim_{\substack{z \rightarrow x \\ z \in S}} f^+(z; y) \text{ exists and equals } f'(x; y).$$

The following lemma encapsulates the heart of our chain rule.

LEMMA 2.4. *Let U be a nonempty open subset of a Banach space X , and let $f \equiv (f_1, f_2, \dots, f_m)$ be a locally Lipschitz mapping from U into R^m . Furthermore, let g be a real-valued locally Lipschitz function defined on a nonempty open subset V of R^m which contains $f(U)$.*

If f is strictly differentiable at some point $x_0 \in U$, in the direction y , and $g^0(f(x_0); f'(x_0; y)) = -g^0(f(x_0); -f'(x_0; y))$, then $g \circ f$ is strictly differentiable at x_0 , in the direction y .

Proof. Let $D \equiv \{x \in U : (g \circ f)'(x; y) = g'(f(x); f'(x; y))\}$. It follows from Remark 2.2 that D is $1 - D$ almost everywhere in U , in the direction y .

(i) Let us first observe that $x_0 \in D$. From Remark 2.3 we see that to show $(g \circ f)^0(x_0; y) = -(g \circ f)^0(x_0; -y)$ we need only show that

$$\lim_{\substack{z \rightarrow x_0 \\ z \in D}} (g \circ f)'(z; y) \text{ exists and equals } (g \circ f)'(x_0; y).$$

So suppose $\epsilon > 0$. By the continuity of the mapping, $x \rightarrow g^+(x; f'(x_0; y))$, at $f(x_0)$ there exists an open neighborhood W_1 of x_0 such that

$$|g^+(f(z); f'(x_0; y)) - g^+(f(x_0); f'(x_0; y))| < \epsilon/2$$

for each $z \in W_1$. Now, as g is locally Lipschitz there exists an open neighborhood W_2 of x_0 and a constant $M > 0$ such that

$$|g'(f(z); f'(z; y)) - g^+(f(z); f'(x_0; y))| \leq M \|f'(z; y) - f'(x_0; y)\|$$

for each $z \in D \cap W_2$. On the other hand, $x \rightarrow f^+(x; y)$ is continuous at x_0 ; therefore, there exists an open neighborhood W_3 of x_0 such that

$$\|f'(x_0; y) - f^+(x; y)\| < \epsilon/2M$$

for each $x \in W_3$. Hence we have that for each $x \in D \cap W_1 \cap W_2 \cap W_3$,

$$\begin{aligned} |(g \circ f)'(x_0; y) - (g \circ f)'(x; y)| &= |g'(f(x_0); f'(x_0; y)) - g'(f(x); f'(x; y))| \\ &\leq |g'(f(x_0); f'(x_0; y)) - g^+(f(x); f'(x_0; y))| \\ &\quad + |g^+(f(x); f'(x_0; y)) - g'(f(x); f'(x; y))| \\ &\leq \epsilon/2 + \epsilon/2 = \epsilon. \end{aligned}$$

Therefore,

$$\lim_{\substack{z \rightarrow x_0 \\ z \in D}} (g \circ f)'(z; y) = (g \circ f)'(x_0; y). \quad \square$$

We need a few more definitions before we can show that $\mathcal{A}_e(U)$ contains a significant class of functions (above and beyond the C^1 functions). Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a Banach space X . Then f is *upper semismooth* (*lower semismooth*) at a point $x \in U$, in the direction y , if

$$f^+(x; y) \geq \limsup_{\substack{t \rightarrow 0^+ \\ y' \rightarrow y}} f^+(x + ty'; y) \quad (f^-(x; y) \leq \liminf_{\substack{t \rightarrow 0^+ \\ y' \rightarrow y}} f^-(x + ty'; y)).$$

Moreover, we say that f is *semismooth* at a point $x \in U$, in the direction y , if

$$\limsup_{\substack{t \rightarrow 0^+ \\ y' \rightarrow y}} f^+(x + ty'; y) = f^+(x; y) = f^-(x; y) = \liminf_{\substack{t \rightarrow 0^+ \\ y' \rightarrow y}} f^-(x + ty'; y).$$

To see that this terminology is justified we direct the reader's attention to Corollary 6.3 in [7]. If X is finite dimensional then we say that f is *arcwise essentially upper semismooth* (*arcwise essentially lower semismooth*) on U if for each $x \in \mathcal{S}_e((0, 1); U)$ we have that f is upper (lower) semismooth at $x(t)$ in the direction $x'(t)$ for almost all $t \in (0, 1)$.

The following technical result allows us to perform only "unilateral" analysis in our main theorem.

THEOREM 2.1. *Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a finite dimensional Banach space X . Then f is arcwise essentially upper semismooth (arcwise essentially lower semismooth) on U if and only if $f \in \mathcal{A}_e(U)$.*

Proof. Suppose that f is arcwise essentially upper semismooth on U (the proof for the case when f is arcwise essentially lower semismooth on U is similar). Let $x \in \mathcal{S}_e((0, 1); U)$. Consider the mappings $P : (0, 1) \rightarrow R$ and $T : (0, 1) \rightarrow R$ defined by

$$P(t) \equiv f^0(x(t); x'(t)) \quad \text{and} \quad T(t) \equiv \limsup_{t' \rightarrow t^+} P(t').$$

Clearly, $T(t) \leq P(t)$ almost everywhere in $(0, 1)$. Indeed, $T(t) \leq P(t)$ at each point of $S \equiv \{t \in (0, 1) : x \text{ is strictly differentiable at } t\}$. We claim that $E \equiv \{t \in S : T(t) < P(t)\}$ is countable (and hence $T(t) = P(t)$ almost everywhere in $(0, 1)$). Without loss of generality let us assume that $E \neq \emptyset$. In this case, we define $s : E \rightarrow Q^2$ by $s(t) \equiv (r_1, r_2)$, where $r_1 \in (T(t), P(t)) \cap Q$ and $r_2 \in (t, 1) \cap Q$ is chosen so that $\sup\{P(t') : t < t' < r_2\} < r_1$. It is easy to see that s is 1-to-1 and so E must be at most countable (here Q denotes the rational numbers). We now show that $P(t) = f^+(x(t); x'(t))$ at almost every point of $S \setminus E$. To prove this, consider an arbitrary point $t_0 \in S \setminus E$ where f is upper semismooth at $x(t)$ in the direction $x'(t)$. Let $\epsilon > 0$ be fixed but arbitrary. We will show that $T(t_0) \leq f^+(x(t_0); x'(t_0)) + \epsilon$. Note that without loss of generality we may assume that $x'(t_0) \neq 0$. Now, since f is upper semismooth at $x(t_0)$ in the direction $x'(t_0)$, there exists a $0 < \delta_1$ such that

$$f^+(z; x'(t_0)) \leq f^+(x(t_0); x'(t_0)) + \epsilon/2$$

for each z in the nonempty open subset V , where V equals

$$V \equiv \{z \in U : z = x(t_0) + ty, \ 0 < t < \delta_1 \text{ and } \|y - x'(t_0)\| < \delta_1\}.$$

Let

$$E(\lambda) \equiv \frac{x(t_0 + \lambda) - x(t_0)}{\lambda} - x'(t_0) \quad \text{for } 0 < \lambda < (1 - t_0).$$

Since x is Gâteaux differentiable at t_0 , there exists a $\delta_2 \in (0, \delta_1)$ such that $\|E(\lambda)\|_\infty < \delta_1$ for each $\lambda \in (0, \delta_2)$. Therefore, for each $\lambda \in (0, \delta_2)$

$$x(t_0 + \lambda) = x(t_0) + \lambda(x'(t_0) + E(\lambda)) \in V,$$

and so, $f^0(x(t_0 + \lambda); x'(t_0)) \leq f^+(x(t_0); x'(t_0)) + \epsilon/2$. Now, as f is locally Lipschitz there exists a constant $M > 0$ and a $\delta_3 \in (0, \delta_2)$ such that

$$|P(t_0 + \lambda) - f^0(x(t_0 + \lambda); x'(t_0))| \leq M\|x^+(t_0 + \lambda) - x'(t_0)\| < \epsilon/2$$

for each $\lambda \in (0, \delta_3)$. From this we get that $P(t_0 + \lambda) \leq f^+(x(t_0); x'(t_0)) + \epsilon$ for each $\lambda \in (0, \delta_3)$, and so, $T(t_0) \leq f^+(x(t_0); x'(t_0)) + \epsilon$. However, since ϵ was arbitrary, $P(t_0) = T(t_0) \leq f^+(x(t_0); x'(t_0))$. The result now follows from Lemma 2.1. The converse is obvious. $\square$

Next we show that the family of functions $\mathcal{A}_e(U)$ enjoys quite remarkable closure properties.

THEOREM 2.2. *Let U be a nonempty open subset of R^n , and suppose that $f_1, f_2, \dots, f_m \in \mathcal{A}_e(U)$. Let g be a real-valued locally Lipschitz function defined on a nonempty open subset V of R^m , which contains $f(U)$, where $f \equiv (f_1, f_2, \dots, f_m)$. Then $g \circ f \in \mathcal{A}_e(U)$ whenever $g \in \mathcal{A}_e(V)$.*

Proof. Let $x \in \mathcal{S}_e((0, 1); U)$, and define $z : (0, 1) \rightarrow V$ by $z(t) \equiv f(x(t))$. Our first task is to show that $z \in \mathcal{S}_e((0, 1); V)$. However, this follows immediately from Lemma 2.4 and the fact that $f \in \mathcal{A}_e(U)$.

Let $S_f \equiv \{t \in (0, 1) : f_j^0(x(t); x'(t)) = -f_j^0(x(t); -x'(t)) \text{ for } 1 \leq j \leq m\}$ and $S_g \equiv \{t \in (0, 1) : g^0(z(t); z'(t)) = -g^0(z(t); -z'(t))\}$. Now, define $S \equiv S_f \cap S_g$. Clearly, $\mu(S) = 1$.

We claim that $(g \circ f)^0(x(t); x'(t)) = -(g \circ f)^0(x(t); -x'(t))$ at each point of S . To this end, consider any point $t_0 \in S$. Let us first observe that if $x'(t_0) = 0$, then we are done. So suppose that $x'(t_0) \neq 0$. Set $x_0 \equiv x(t_0)$ and $y \equiv x'(t_0)$. Then we have that f is strictly differentiable at x_0 , in the direction y , and $g^0(f(x_0); f'(x_0; y)) = -g^0(f(x_0); -f'(x_0; y))$ since $t_0 \in S_g$.

Therefore from Lemma 2.4, $g \circ f$ is strictly differentiable at x_0 , in the direction y ; that is, $(g \circ f)^0(x(t_0); x'(t_0)) = -(g \circ f)^0(x(t_0); -x'(t_0))$. $\square$

On relaxing the hypotheses on $f_1, f_2, \dots, f_m$ we may still recover a satisfactory and important theorem.

THEOREM 2.3. *Let U be a nonempty open subset of a Banach space X , and let $f \equiv (f_1, f_2, \dots, f_m) \in \mathcal{S}_e(U; V)$, where V is any nonempty open subset of R^m that contains $f(U)$. Then $g \circ f \in \mathcal{S}_e(U)$ whenever $g \in \mathcal{A}_e(V)$.*

Proof. It suffices to show that for each $y \in S(X)$, $(g \circ f)^0(x; y) = -(g \circ f)^0(x; -y)$ almost everywhere in U . So fix $y \in S(X)$. Let D be the (G_δ) set of all points in U where $f_j^0(x; y) = -f_j^0(x; -y)$ for each $1 \leq j \leq m$, and let $P_y \equiv \{x \in U : (g \circ f)^0(x; y) = -(g \circ f)^0(x; -y)\}$. Clearly P_y is a Borel set; in fact, P_y is a G_δ set.

Let H be any closed hyperplane in X such that $y \notin H$. Now consider the isomorphism $T : H \times R \rightarrow X$ defined by $T(h, t) \equiv h + ty$. Let

$$H_D \equiv \{h \in H : \mu(\{t \in R : T(h, t) \in U \setminus D\}) = 0\}.$$

By Theorem 2.3 in [4] we see that $H \setminus H_D$ is a Haar-null set in H , since $U \setminus D$ is a Haar-null set in X . To show that $U \setminus P_y$ is a Haar-null set in X it suffices (also because of Theorem 2.3 in [4]) to show that for each $h \in H_D$, $\mu(\{t \in R : T(h, t) \in U \setminus P_y\}) = 0$.

To this end, consider $h_0 \in H_D$. Let $U_{h_0} \equiv \{t \in R : T(h_0, t) \in U\}$. If $U_{h_0} = \emptyset$ then we are done, so let us suppose that $U_{h_0} \neq \emptyset$. Define $z : U_{h_0} \rightarrow V$ by $z(t) \equiv f(h_0 + ty)$. Since $h_0 \in H_D$, $z \in \mathcal{S}_e(U_{h_0}; V)$. Let

$$D_f \equiv \{t \in U_{h_0} : f_j^0(h_0 + ty; y) = -f_j^0(h_0 + ty; -y) \text{ for all } 1 \leq j \leq m\}$$

and

$$D_g \equiv \{t \in U_{h_0} : g^0(z(t); z'(t)) = -g^0(z(t); -z'(t))\}.$$

Now define $D_0 \equiv D_f \cap D_g$. Clearly, $\lambda(U_{h_0} \setminus D_0) = 0$. We claim that

$$(g \circ f)^0(h_0 + ty; y) = -(g \circ f)^0(h_0 + ty; -y)$$

at each point $t \in D_0$. To see this, consider an arbitrary point $t_0 \in D_0$. Set $x_0 \equiv h_0 + t_0 y$, then $g^0(f(x_0); f'(x_0; y)) = -g^0(f(x_0); -f'(x_0; y))$ as $t_0 \in D_g$ and $f_j^0(x_0; y) = -f_j^0(x_0; -y)$ for each $1 \leq j \leq m$ as $t_0 \in D_f$. It now follows from Lemma 2.4 that $(g \circ f)^0(x_0; y) = -(g \circ f)^0(x_0; -y)$. This completes the proof. $\square$

Remark 2.4. It follows from Theorem 2.3 that the distance function g defined in Example 1.1 lies in $\mathcal{S}_e(R^m) \setminus \mathcal{A}_e(R^m)$, ($m \geq 2$). However, by translation and dilation (of this example) one can show that for every nonempty open subset U of R^m ($m \geq 2$), $\mathcal{A}_e(U)$ is a proper subset of $\mathcal{S}_e(U)$.

COROLLARY 2.1. *Let U be a nonempty open subset of R^m . Then $\mathcal{A}_e(U)$ and $\mathcal{S}_e(U)$ are function algebras and vector lattices containing all the convex and C^1 functions.*

Let us also observe that the class of functions $\mathcal{A}_e(R^m)$ contains many distance functions. Indeed, one can show that if C is a closed regular subset of R^m [6], then the distance function generated by this set and any smooth norm on R^m is in this class. To see this, we need to make the following observations:

- (i) For any norm on R^m , the distance function d_C is regular at each point of C , see [1] or [6].
- (ii) If the norm is smooth on R^m , then $-d_C$ is regular on $R^n \setminus C$, see [2].

We can now define a class of closed subsets of R^m as follows. A closed subset C of R^m is *arcwise essentially smooth* if for each $x \in \mathcal{S}_e((0, 1); R^m)$ the set

$$\{t \in (0, 1) : x(t) \in C \text{ and } x'(t) \in K_C(x(t)) \setminus T_C(x(t))\}$$

has measure zero. Here $K_C(x)$ denotes the *contingent cone* of C at x and $T_C(x)$ denotes the *Clarke tangent cone* of C at x . Recall that C is regular at x if $K_C(x) = T_C(x)$ [6]. Note that all sets that are regular except on a countable set are arcwise essentially smooth. Thus, all closed convex sets and all smooth manifolds are arcwise essentially smooth.

PROPOSITION 2.3. *Let $\|\cdot\|$ be a smooth norm on R^m , and let C be a nonempty closed subset of R^m . Then the distance function generated by the norm $\|\cdot\|$ and the set C is a member of $\mathcal{A}_e(R^m)$ if and only if C is arcwise essentially smooth.*

It is immediate from this proposition and Theorem 2.2 that, unlike the family of regular sets, the family of arcwise essentially smooth sets is closed under finite unions. We should also observe that from Proposition 2.1 it follows that the boundary of every arcwise essentially smooth set is a Lebesgue-null set. We also note that the set $\{0\} \times \{0\} \times \cdots \times C$ in Example 1.1 is an example of a null set that is not arcwise essentially smooth. Finally, observe that we may now provide general conditions to ensure that exact penalty functions of the form

$$f(x) + Kd_C(x)$$

will be in $\mathcal{A}_e(U)$ or in $\mathcal{S}_e(U)$.

REFERENCES

- [1] J. M. BORWEIN AND M. FABIAN, *A note on regularity of sets and of distance functions in Banach space*, J. Math. Anal. Appl., 182 (1994), pp. 566–560.
- [2] J. M. BORWEIN, S. P. FITZPATRICK, AND J. R. GILES, *The differentiability of real functions on normed linear spaces using generalized subgradients*, J. Math. Anal. Appl., 128 (1987), pp. 512–534.
- [3] J. M. BORWEIN AND W. B. MOORS, *Essentially smooth Lipschitz functions*, J. Funct. Anal., 149 (1997), pp. 305–351.
- [4] J. M. BORWEIN AND W. B. MOORS, *Null sets and essentially smooth Lipschitz functions*, SIAM J. Optim., 8 (1998), pp. 309–323.
- [5] J. P. R. CHRISTENSEN, *On sets of Haar measure zero in Abelian groups*, Israel J. Math., 13 (1972), pp. 255–260.
- [6] F. H. CLARKE, *Optimization and Nonsmooth Analysis*, John Wiley, New York, 1971.
- [7] R. CORREA AND A. JOFRE, *Tangentially continuous directional derivatives in nonsmooth analysis*, JOTA, 61 (1989), pp. 1–21.
- [8] W. B. MOORS, *A characterisation of minimal subdifferential mappings of locally Lipschitz functions*, Set-Valued Anal., 3 (1995), pp. 129–141.
- [9] M. VALADIER, *Entrainement unilatéral, lignes de descente, fonctions Lipschitziennes non pathologiques*, C. R. Acad. Sci. Paris 308, Series I, 1989, pp. 241–244.

NULL SETS AND ESSENTIALLY SMOOTH LIPSCHITZ FUNCTIONS*

JONATHAN M. BORWEIN[†] AND WARREN B. MOORS[‡]

Abstract. In this paper we extend the notion of a Lebesgue-null set to a notion which is valid in any completely metrizable Abelian topological group. We then use this definition to introduce and study the class of essentially smooth functions. These are, roughly speaking, those Lipschitz functions which are smooth (in each direction) almost everywhere.

Key words. Lipschitz functions, Haar-null sets, essentially smooth functions

AMS subject classifications. Primary, 49J520, 46N10; Secondary, 58C20

PII. S1052623496305213

1. Introduction. The primary purpose of this paper is to introduce a new class of Lipschitz functions and present some of their most fundamental properties. In order to accomplish this goal, we must first extend the notion of a Lebesgue-null set to a notion which is valid in any Banach space. In the case of separable Banach spaces this task has already been achieved. In fact, for separable Banach spaces, there have been several successful solutions to this problem (see [20, 15, 7], to name just a few). For our purposes the most useful generalization of a “null” set is that of the Haar-null (or zero) set introduced by J. P. R. Christensen in [7]. Indeed, in section 2 of this paper we retrace and extend the results of Christensen to spaces which are not necessarily separable. While some of our extensions need additional work, many follow naturally from their separable counterparts.

In section 3 we introduce the class of “essentially smooth” Lipschitz functions. Loosely speaking, these are those functions which are smooth (in each direction) almost everywhere, that is, smooth (in each direction) everywhere except on a “null” set. As we shall come to see, the essentially smooth functions enjoy particularly desirable differentiability properties. For instance, they are all integrable and D -representable (on class($\mathcal{S}$) spaces) in the sense defined within.

This class of functions is of central importance in nonsmooth optimization. The importance of the class stems (see [2, 4, 5, 6]) from the fact that it provably provides the largest robust class of functions

- stable under the most significant operations of calculus and of formation of marginal functions [4];
- uniquely determined by their Clarke subgradients [4];
- smooth enough for algorithmic purposes [4, 18];
- possessing strong chain rules [5].

A primary utility of this paper is to provide coherent tools for the use of “measure-theoretic” techniques in nonseparable spaces, such as L^∞ , which are frequently natural settings for optimization and control problems.

*Received by the editors June 12, 1996; accepted for publication (in revised form) January 21, 1997.

<http://www.siam.org/journals/siopt/8-2/30521.html>

[†]CECM, Department of Mathematics and Statistics, Simon Fraser University, Burnaby, BC V5A 1S6, Canada (jborwein@cecm.sfu.ca). This author was supported by NSERC and the Shrum Endowment at Simon Fraser University.

[‡]Department of Mathematics, The University of Auckland, Private Bag 92019, Auckland, New Zealand (moors@math.auckland.ac.nz). This author was supported by a New Zealand Science and Technology Post-Doctoral Fellowship.

2. Haar-null sets. Throughout this section of the paper, $(G, +, \tau)$ will denote a completely metrizable Abelian topological group; that is, (G, τ) is homeomorphic to a complete metric space. Due to a result of V. Klee (see [14]), we may and do assume that the topology τ is generated by a metric d , which is both complete and invariant.

In our terminology a *measure* will always be nonnegative and not identical to zero. By a *Borel measure* on a topological space X we mean any measure defined on $\mathcal{B}(X)$ —the Borel subsets of X . By a *Radon measure* μ on X we mean any Borel measure on X extended to its completion on X , which satisfies

- (i) $\mu(K) < \infty$ for each compact subset $K \subseteq X$;
- (ii) $\mu(A) = \sup\{\mu(K) : K \subseteq A, K \text{ compact}\}$ for each $A \in \mathcal{B}(X)$.

Note that (ii) actually holds for each A in the μ -completion of $\mathcal{B}(X)$. In addition, if μ is finite (that is, $\mu(X) < \infty$), then $\mu(A) = \inf\{\mu(U) : A \subseteq U, U \text{ open}\}$ for each A in the μ -completion of $\mathcal{B}(X)$.

We will say that a subset A of a topological space X is *universally (Radon) measurable* if it belongs to the μ -completion of each finite (Radon) Borel measure μ on X , and we shall denote by $\mathcal{U}(X)$ ($\mathcal{U}_R(X)$) the family of all universally (Radon) measurable subsets on X . Observe that $\mathcal{B}(X) \subseteq \mathcal{U}(X) \subseteq \mathcal{U}_R(X)$. (In the case when X is Polish, $\mathcal{U}(X) = \mathcal{U}_R(X)$; see Theorem 2.2.) It follows immediately from this definition that on any topological group the universally (Radon) measurable sets form a translation invariant σ -algebra.

A natural question to ask at this point is whether the family of all universally measurable subsets properly contains the Borel sets. One way to see the affirmative answer to this is the following.

Given a completely regular topological space X , let $\mathcal{D}(X)$ denote the smallest σ -algebra on X which contains all the Borel subsets and is stable under the Souslin operation. (Let us refresh our memory on the definition of the Souslin operation. We say that a family of sets $\mathcal{A}$ is *closed under the Souslin operation* if all sets of the form

$$\bigcup \left\{ \bigcap \{A(\sigma|n) : n \geq 1\} : \sigma \in N^N \right\}, \quad A(\sigma|n) \in \mathcal{A}$$

(here we use $\sigma|n$ to denote $\sigma_1, \sigma_2, \dots, \sigma_n$ when $(\sigma_1, \sigma_2, \dots, \sigma_n, \dots) \in N^N$)

are contained in $\mathcal{A}$.) By using the fact that for each complete finite measure μ on X the family of all μ -measurable sets is closed under the Souslin operation, we may deduce that each member of $\mathcal{D}(X)$ is universally measurable (see [21]). Now, it is known that $\mathcal{D}(X)$ contains all the Čech-analytic subsets of X , which, in turn, contain all the analytic subsets of X (see [12]). (Recall that a subset $A \subseteq X$ is *analytic* if it is the continuous image of a complete, separable metric space; that is, the continuous image of a *Polish* space.) Finally, it is also known that each uncountable Polish space contains an analytic set which is not a Borel set (see [13, p. 201]).

We are now in a position to define our “null” sets. Let $(G, +, \tau)$ be a completely metrizable Abelian topological group. We will call a universally Radon measurable set $A \subseteq G$ a *Haar-null* set if there exists a (not necessarily unique) Radon probability measure p on G such that $p(g + A) = 0$ for each $g \in G$. (In such a case, we shall call the measure p a *test measure* for A .) More generally, we will say that a subset $A \subseteq G$ is a Haar-null set if it is contained in a universally Radon measurable Haar-null set. It will sometimes be useful to remember that if p is a test measure for A , then $g + A$ is p -measurable for each $g \in G$, even though the set A may not be universally Radon measurable. We should also note that the Dirac (point-mass) measures are rarely useful as test measures since the only set they can *test* is the empty set.

In order to simplify the later part of this section, we will take this opportunity to present some of the fundamental properties possessed by Radon measures. All of the results presented in the next theorem are either straightforward or may be found (more or less as stated) in Chapter 2 of [1].

THEOREM 2.1. *Let X and Y be Hausdorff topological spaces, and let Z be a Borel subset of X . Further, let μ be a Radon measure on X and ν be a Radon measure on Y . Then we have the following:*

(a) *The space X contains a smallest closed subset $A \subseteq X$ (called the support of μ) such that $\mu(X \setminus A) = 0$. It follows immediately from this definition that, for each open subset $U \subseteq X$, $\mu(U \cap A) > 0$ whenever $U \cap A \neq \emptyset$. Furthermore, if X is metrizable and $\mu(X) < \infty$, then the support of μ is separable.*

(b) *If we define $\mu_Z : \mathcal{B}(Z) \rightarrow [0, \infty]$ by $\mu_Z(A) \equiv \mu(A)$ for each $A \in \mathcal{B}(Z)$, then (the completion of) μ_Z defines a Radon measure on Z , which we will call the restriction of μ .*

(c) *Conversely, if we are given a finite Radon measure μ_Z on Z and we define $\eta : \mathcal{B}(X) \rightarrow [0, \infty]$ by $\eta(A) \equiv \mu_Z(A \cap Z)$ for each $A \in \mathcal{B}(X)$ then (the completion of) η is a Radon measure on X , which in the future we will call the extension of μ_Z . (Note that if Z is a closed subset of X , then we need not assume that μ_Z is finite.)*

(d) $\mathcal{U}_R(Z) = \{A \in \mathcal{U}_R(X) : A \subseteq Z\}$.

(e) *If μ is σ -finite, that is, $X = \bigcup \{X_n : n \in \mathbb{N}\}$ where each $X_n \in \mathcal{B}(X)$ and $\mu(X_n) < \infty$, then each member of $\mathcal{U}_R(X)$ is μ -measurable.*

(f) *If A is a universally Radon measurable subset of $X \times Y$ and both the measures μ and ν are σ -finite, then the mapping $x \rightarrow \nu([A]_X(x))$ is universally Radon measurable on X and $y \rightarrow \mu([A]_Y(y))$ is universally Radon measurable on Y . (Here, $[A]_X(x) \equiv \{y \in Y : (x, y) \in A\}$ and $[A]_Y(y) \equiv \{x \in X : (x, y) \in A\}$.) Moreover, there exists a unique Radon product measure on $X \times Y$, denoted $\mu \otimes \nu$, that restricts to the usual product measure on $\mathcal{B}(X) \times \mathcal{B}(Y)$ and for which*

$$\int_A d(\mu \otimes \nu) = \int_X \nu([A]_X(x)) d\mu = \int_Y \mu([A]_Y(y)) d\nu.$$

(g) *If f is a continuous mapping from X into Y and $\mu(X) < \infty$, then the measure $\mu_f : \mathcal{B}(Y) \rightarrow [0, \infty]$ defined by $\mu_f(A) \equiv \mu(f^{-1}(A))$ extends to define a Radon measure on Y . Moreover, $f^{-1}(A) \in \mathcal{U}_R(X)$ for each $A \in \mathcal{U}_R(Y)$.*

(h) *If μ and ν are finite measures and $X = Y$, then we may define the convolution of μ and ν as $\mu * \nu \equiv (\mu \otimes \nu)_+$; that is,*

$$\mu * \nu(A) \equiv \mu \otimes \nu(\{(x, y) : x + y \in A\}) \text{ for each } A \in \mathcal{B}(X).$$

The completion of this measure is a finite Radon measure on X .

The convolution defined above plays a central role in what follows.

Remark 2.1. It follows from (f) and (h) above that if p_A is a test measure for a set A and p_B is a test measure for a set B , then $p_A * p_B$ is a test measure for $A \cup B$. Indeed, since $p_A * p_B(A \cup B) \leq p_A * p_B(A) + p_A * p_B(B)$, it suffices to observe that $p_A * p_B(A) = 0$ and $p_A * p_B(B) = 0$. As a consequence of this, we may deduce that the Haar-null sets are closed under finite unions. Later, we shall see that this remains true of countable unions.

Before we present the elementary properties possessed by Haar-null sets, we will digress for a moment so that we may justify calling our “null” sets Haar-null sets, as in [7].

THEOREM 2.2 (Ulam's theorem [9, p. 176]). *Let (X, τ) be a separable and completely metrizable topological space (that is, X is a Polish space). Then the completion of each finite Borel measure on X defines a finite Radon measure on X .*

A generalization of the above theorem is the following.

Remark 2.2 If the support of a finite Borel measure is Polish, then its completion is a finite Radon measure. In particular, if (X, τ) is metrizable, then each finite Borel measure on X , with compact support, extends to be a finite Radon measure on X .

So we see then that the definition of a Haar-null set given in this paper is consistent with that given in [7]. Moreover, it also coincides with the definition of a *shy* set, as given in [11].

A key set of facts is listed next.

PROPOSITION 2.3. *Let $(G, +, \tau)$ be a completely metrizable Abelian group. Then we have the following:*

- (a) *Every subset of a Haar-null set is Haar-null.*
- (b) *If A is a Haar-null set, then so is $g + A$ for every $g \in G$.*
- (c) *If A is a Haar-null set, then there exists a test measure for A with compact support.*
- (d) *Suppose that A is a subset of a universally Radon measurable set B . Then to show that A is a Haar-null set, it is sufficient to show that there exists a test measure for B which is σ -finite. Note that if G is Polish, then all Radon measures on G are σ -finite (see [1, p. 62]), and so in this case it is sufficient to use any Radon measure. If B is a Borel set, then we may use any Borel measure μ (as a test measure) for which there exists a compact subset $K \subseteq G$ with $0 < \mu(K) < \infty$.*
- (e) *If A is a Haar-null set, then $G \setminus A$ is dense in G .*
- (f) *If $\{A_j : j \in N\}$ are Haar-null sets, then so is $\bigcup \{A_j : j \in N\}$.*
- (g) *If G is a locally compact Polish group, then a subset $A \subseteq G$ is a Haar-null set if and only if $\lambda(A) = 0$ for the Haar-measure λ on G .*

Proof. The proof of the facts (a)–(c) are obvious and hence left to the reader.

(d) Suppose that μ is a test measure for the set B . Let K be any compact subset of G such that $0 < \mu(K) < \infty$. It is easy to see that the finite Radon measure, μ_K , defined by $\mu_K(X) \equiv \mu(K \cap X)$ for each $X \in \mathcal{B}(G)$ is also a test measure for the set B .

To prove (e) it is sufficient to show that there are no (nonempty) open Haar-null sets. To this end, let U be a nonempty open subset of G and suppose, to the contrary, that there does exist a test measure p for U on G . Let A denote the support of p . It is easy to see that for some $g_0 \in G$, $(g_0 + U) \cap A \neq \emptyset$. Hence, $p(g_0 + U) \geq p((g_0 + U) \cap A) > 0$, which contradicts the fact that p is a test measure for U . Hence, no nonempty open subset of G is a Haar-null set.

(f) Let us first observe that without loss of generality we may assume that each set A_j is universally Radon measurable. For each $j \in N$, let p_j be a test measure for A_j on G . Let H be the smallest closed subgroup of G that contains the support of each p_j . Since the support of each p_j is separable (see Theorem 2.1 (a)) it is not too difficult to see that H is Polish. Next, let p_j^* denote the restriction of p_j to H . We know from Theorem 1 in [7] that there exists a Radon probability measure p^* on H that is a test measure for each set of the form $\bigcup \{B_j : j \in N\}$, provided that $B_j \in \mathcal{U}_R(H)$ and p_j^* is a test measure for B_j . (The measure p^* is essentially the infinite convolution of the p_j^* .) Let p be the extension, to G , of p^* . We claim that p is a test measure for $\bigcup \{A_j : j \in N\}$. To prove this, we must show that for each $g \in G$,

$p(g + \bigcup\{A_j : j \in N\}) = 0$. So let us fix $g \in G$. Then,

$$p\left(g + \bigcup_{j=1}^{\infty} A_j\right) = p^*\left(\left(g + \bigcup_{j=1}^{\infty} A_j\right) \cap H\right) = p^*\left(\bigcup_{j=1}^{\infty} (g + A_j) \cap H\right).$$

However, each p_j^* is a test measure for $(g + A_j) \cap H$ since each p_j is a test measure for A_j , and $h + ((g + A_j) \cap H) = ((h + g) + A_j) \cap H \subseteq (h + g) + A_j$ for each $h \in H$. Therefore, p^* is a test measure for $\bigcup\{(g + A_j) \cap H : j \in N\}$, and so $p^*(\bigcup\{(g + A_j) \cap H : j \in N\}) = 0$, which gives that $p(g + \bigcup\{A_j : j \in N\}) = 0$.

Let us also note that an alternative proof of this is given in [11]. We should, however, lend a note of caution here that in their proof they *implicitly* assume that the metric under consideration is both translation invariant and complete. (Of course, from our remark at the start of this section, we see that such an assumption is indeed valid.)

(g) The proof of this fact may be found in [7]. $\square$

One of the most powerful theorems in measure theory is Fubini's theorem, and so it is natural to want to determine whether there is a version of Fubini's theorem that holds for Haar-null sets. Disappointingly, in [7] the author gives an example to show that we must banish all hope of obtaining a *full* version of Fubini's theorem. Nonetheless, in this same paper the author indicates (without proof) that a weaker version of Fubini's theorem does hold (in a Polish Abelian group). We next take the opportunity to record a proof of this theorem in the setting of completely metrizable Abelian groups.

THEOREM 2.4. *If $(H, +, \tau_1)$ is a completely metrizable Abelian group and $(T, +, \tau_2)$ is a locally compact Polish Abelian group, then for each universally Radon measurable subset $A \subseteq H \times T$ (the product group) the following are equivalent:*

- (i) $[A]_H(h)$ is a Haar-null set, for the Haar-measure on T , for almost all $h \in H$.
- (ii) The set A is a Haar-null set in the product group $H \times T$.

Proof. Let p_T be a Radon probability measure on T that is equivalent to the Haar-measure on T ; that is, for each $B \in \mathcal{U}_R(T)$, $p_T(B) = 0$ if and only if B is a null set for the Haar-measure on T . (Note that such a measure p_T exists since T , with the Haar-measure, is σ -finite.)

Suppose that (i) holds. Let $A_H \equiv \{h \in H : \mu([A]_H(h)) > 0\}$. Let p be a test measure for the set A_H . We claim that $p \otimes p_T$ is a test measure for A . Fix $(h_0, t_0) \in H \times T$. Then

$$\{h \in H : p_T([(h_0, t_0) + A]_H(h)) > 0\} = h_0 + A_H.$$

The result now follows from Theorem 2.1 (f).

Suppose that (ii) holds. Let δ_0 denote the Dirac (point-mass) measure at 0_H and define $\mu \equiv \delta_0 \otimes p_T$. Let p be a test measure for the set A and set $A_H \equiv \{h \in H : p_T([A]_H(h)) > 0\}$. We know from Theorem 2.1 (f) that A_H is universally Radon measurable, so it suffices to construct a test measure for A_H . To this end, we define a Radon measure p_H on H by $p_H(B) \equiv p(B \times T)$ for each $B \in \mathcal{U}_R(H)$. (Note that $p_H(B) \equiv p(P^{-1}(B))$, where P is the natural projection of $H \times T$ onto H .) Fix $h_0 \in H$ then

$$0 = \mu * p((h_0, 0_T) + A) = \int_{H \times T} \mu(A + (h_0 - h, -t)) dp(h, t).$$

Now, the mapping $(h, t) \rightarrow \mu(A + (h_0 - h, -t)) = p_T([A + (0_H, -t)]_H(h - h_0))$ is universally Radon measurable and hence p -measurable (see Theorem 2.1 (f) and (g)) and strictly positive on $(h_0 + A_H) \times T$.

Hence, $p_H(h_0 + A_H) = p((h_0 + A_H) \times T) = 0$. $\square$

There is one significant problem which, up until this point, we have neglected. That is, how does one actually determine whether a given set is Haar-null? In general, this is a difficult question to answer. In some situations, however, the following facts are helpful.

PROPOSITION 2.5. *Let A be a Borel subset of a completely metrizable Abelian group $(G, +, \tau)$. Then A is a Haar-null set if for some locally compact subgroup H of G , $\lambda((g + A) \cap H) = 0$ for each g in some transversal T to H in G . Here, λ denotes the Haar-measure on H .*

Let us recall the definition of a transversal. A subset T of a group G , with normal subgroup H , which contains just one element from each coset of H is called a *transversal to H in G* .

The next condition is stated in [7] for Polish groups. However, we note here that the proof in no way relies on the fact that the group is separable.

PROPOSITION 2.6. *Let A and B be arbitrary universally Radon measurable sets in a completely metrizable Abelian group G . If we put*

$$F(A, B) \equiv \{g \in G : (g + A) \cap B \text{ is not a Haar-null set}\},$$

then $F(A, B)$ is an open subset of G (possibly empty). In particular, if A is not a Haar-null set, then $F(A, A)$ is an open neighborhood of 0_G .

COROLLARY 2.7. *If $(G, +, \tau)$ is a completely metrizable Abelian group which is not locally compact, then each compact subset is Haar-null.*

Proof. Suppose that, to the contrary, there exists a compact subset $K \subseteq G$ which is not a Haar-null set. Then, by Proposition 2.6, $F(K, K)$ is an open neighborhood of 0_G . However, $0_G \in F(K, K) \subseteq K - K$, which is compact. Therefore, all compact subsets of G must be Haar-null. $\square$

By employing considerably more effort, one can show, in the setting of Banach spaces, that certain more complicated sets are Haar-null. For instance, in [3] the authors show that the boundary of every closed convex subset of every separable super-reflexive Banach space is Haar-null. Moreover, it is shown in [16] that the boundary of every closed convex nowhere-dense set in *any* super-reflexive space is Haar-null. (Note that in [16] the author does not use the Haar-null terminology.) It is known (see [17]) that every separable nonreflexive Banach space contains a closed convex subset whose boundary is not Haar-null. (Note that such a set must have no interior.)

We end this section with an example that shows that the completeness of the group G in the proof of Proposition 2.3 part (f) is indispensable.

Example 2.1. Let $(X, \|\cdot\|)$ be an infinite dimensional normed linear space such that $X = \bigcup\{X_n : n \in N\}$, where each X_n is a finite dimensional subspace of X . It follows from Proposition 2.5 that each subspace X_n is a Haar-null subset of X . (To see this, consider the following. For each $n \in N$, let H_n be a one-dimensional subspace of X such that $H_n \not\subseteq X_n$ and let λ_n denote the Lebesgue measure on H_n , then $\lambda_n((x + X_n) \cap H_n) = 0$ for each $x \in X$.) However, $p(\bigcup\{X_n : n \in N\}) = 1$ for each probability measure p on X .

3. Essentially smooth Lipschitz functions. In this section of the paper we will introduce the class of essentially smooth Lipschitz functions and develop some of

their most fundamental properties. We begin by recalling some preliminary definitions regarding the Clarke generalized directional derivative [8]. A real-valued function f defined on a nonempty open subset U of a Banach space X is said to be *locally Lipschitz* on U if for each $x_0 \in U$ there exists a $K > 0$ and a $\delta > 0$ such that

$$|f(x) - f(y)| \leq K\|x - y\| \quad \text{for all } x, y \in B(x_0, \delta) \cap U.$$

For functions in this class, we will consider the following three directional derivatives:

(i) The *upper Dini derivative* at $x \in U$, in the direction y , given by

$$f^+(x; y) \equiv \limsup_{\lambda \rightarrow 0^+} \frac{f(x + \lambda y) - f(x)}{\lambda}.$$

(ii) The *lower Dini derivative* at $x \in U$, in the direction y , given by

$$f^-(x; y) \equiv \liminf_{\lambda \rightarrow 0^+} \frac{f(x + \lambda y) - f(x)}{\lambda}.$$

(iii) The *Clarke generalized directional derivative* at $x \in U$, in the direction y , given by

$$f^0(x; y) \equiv \limsup_{\substack{z \rightarrow x \\ \lambda \rightarrow 0^+}} \frac{f(z + \lambda y) - f(z)}{\lambda}.$$

It is immediate from these definitions that for each $x \in U$ and each $y \in X$

$$f^-(x; y) \leq f^+(x; y) \leq f^0(x; y).$$

An initially surprising but important fact regarding the Clarke generalized directional derivative is that for each $x_0 \in U$, the mapping $y \rightarrow f^0(x_0; y)$ is a continuous sublinear functional on X (see [8]). Associated with the Clarke generalized directional derivative is the *Clarke subdifferential mapping*, which is defined by

$$\partial f(x) \equiv \{x^* \in X^* : x^*(y) \leq f^0(x; y) \text{ for each } y \in X\} \text{ for each } x \in U.$$

Next, we shall examine some notions of differentiability. We say that a real-valued locally Lipschitz function f is *differentiable* at x , in the direction y , if

$$f'(x; y) \equiv \lim_{\lambda \rightarrow 0} \frac{f(x + \lambda y) - f(x)}{\lambda} \quad \text{exists,}$$

and we say that f is *Gâteaux differentiable* at x if

$$\nabla f(x)(y) \equiv \lim_{\lambda \rightarrow 0} \frac{f(x + \lambda y) - f(x)}{\lambda} \quad \text{exists}$$

for each $y \in X$ and $\nabla f(x)$ is a continuous linear functional on X .

We shall also be interested in a slightly stronger notion of differentiability. A locally Lipschitz function f is said to be *strictly differentiable* at x , in the direction y , if

$$\lim_{\substack{z \rightarrow x \\ \lambda \rightarrow 0^+}} \frac{f(z + \lambda y) - f(z)}{\lambda} \quad \text{exists,}$$

and we say that f is *strictly differentiable* at x , if f is strictly differentiable at x , in every direction $y \in X$. We recall that a locally Lipschitz function f is strictly differentiable at x , in the direction y , if and only if

$$f^0(x; y) = f'(x; y) = -f^0(x; -y).$$

Our first theorem in this section shows us that the Clarke generalized directional derivative is insensitive to Haar-null sets.

LEMMA 3.1 (Mean-Value theorem). *Let f be a real-valued locally Lipschitz function defined on a nonempty open subset of the real line, which contains the nondegenerate interval $[a, b]$. Then there exists a Borel subset M of $[a, b]$, with positive measure, such that for each $t \in M$, $f'(t)$ exists and*

$$f'(t) \geq \frac{f(b) - f(a)}{b - a}.$$

THEOREM 3.2. *Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a Banach space X . Then for each $x \in U$, $y \in X \setminus \{0\}$, and Haar-null set $N \subseteq X$, we have that*

$$f^0(x; y) = \limsup_{\substack{z \rightarrow x \\ z \in U \setminus N}} f^-(z; y) \quad -f^0(x; -y) = \liminf_{\substack{z \rightarrow x \\ z \in U \setminus N}} f^+(z; y).$$

Proof. Since the proof of the second case is identical to that of the first, we shall only consider the first case. Fix $x \in U$, $y \in X \setminus \{0\}$, and let N be a Haar-null subset of X . (Note that, by possibly making N bigger, we may assume that it is universally Radon measurable.) We begin by observing that

$$\limsup_{\substack{z \rightarrow x \\ z \in U \setminus N}} f^-(z; y) \leq \limsup_{z \rightarrow x} f^+(z; y) = f^0(x; y).$$

So it is sufficient to show that

$$f^0(x; y) \leq \limsup_{\substack{z \rightarrow x \\ z \in U \setminus N}} f^-(z; y).$$

Indeed, to accomplish this, it will be sufficient to show that for each $\epsilon > 0$ and each $\delta > 0$

$$f^0(x; y) - \epsilon < \sup\{f^-(z; y) : \|z - x\| \leq \delta \text{ and } z \in U \setminus N\}.$$

So suppose $\epsilon > 0$ and $\delta > 0$ are given. Note that, by possibly making δ smaller, we may assume that $B(x, \delta) \subseteq U$. Let H be any closed hyperplane in X that does not contain the point y , and consider the isomorphism $T : H \times \mathbb{R} \rightarrow X$ defined by $T(h, r) \equiv h + ry$. If we set $N_H \equiv \{h \in H : \lambda(\{r \in \mathbb{R} : T(h, r) \in N\}) > 0\}$, then by Theorem 2.4 we have that N_H is a Haar-null subset of H .

Now, by the definition of $f^0(x; y)$, there exists a point $z \equiv T(h_z, r_z) \in U$ and a $\lambda \in (0, 1)$ such that $\|(z + \lambda y) - x\| < \delta$, $\|z - x\| < \delta$, and

$$\frac{f(z + \lambda y) - f(z)}{\lambda} > f^0(x; y) - \epsilon.$$

Since f is locally Lipschitz on U , we may, by possibly moving z slightly, assume that $h_z \in H \setminus N_H$. Now, by the Mean-Value theorem there exists a set M of positive measure in the interval $(z, z + \lambda y)$, where $f'(z'; y)$ exists and

$$f'(z'; y) \geq \frac{f(z + \lambda y) - f(z)}{\lambda} > f^0(x; y) - \epsilon.$$

Choose $z_0 \in M \setminus N \subseteq U$ (this is possible since $h_z \notin N_H$). Then $\|z_0 - x\| < \delta$, and so

$$f^0(x; y) - \epsilon < f'(z_0; y) \leq \sup\{f^-(z; y) : \|z - x\| < \delta \text{ and } z \in U \setminus N\}.$$

This completes the proof. $\square$

We now isolate the class of essentially smooth functions. A real-valued locally Lipschitz function f defined on a nonempty open subset U of a Banach space X is called *essentially smooth* on U if for each $y \in X \setminus \{0\}$

$$\{x \in U : f^0(x; y) \neq -f^0(x; -y)\} \text{ is a Haar-null subset of } X.$$

Furthermore, we shall denote by $\mathcal{S}_e(U)$ the family of all real-valued essentially smooth locally Lipschitz functions on U .

Our first observation concerning these functions is that on separable Banach spaces they are just those locally Lipschitz functions which are strictly differentiable almost everywhere. (Note that in [2, 6, 19] such functions were called *essentially strictly differentiable*.)

THEOREM 3.3. *Let U be a nonempty open subset of a separable Banach space X . Then each member of $\mathcal{S}_e(U)$ is strictly differentiable almost everywhere in U .*

Proof. Let $\{y_n : n \in N\}$ be a dense subset of $X \setminus \{0\}$. For each $n \in N$, let $N_n \equiv \{x \in U : f^0(x, y_n) \neq -f^0(x, -y_n)\}$. We claim that f is strictly differentiable at each point of $U \setminus \bigcup\{N_n : n \in N\}$. To see this, consider a fixed point $x_0 \in U \setminus \bigcup\{N_n : n \in N\}$. Now, as both mappings $y \rightarrow f^0(x_0; y)$ and $y \rightarrow -f^0(x_0; -y)$ are continuous on X and $f^0(x_0; y_n) = -f^0(x_0; -y_n)$ for each $n \in N$, we must have that $f^0(x_0; y) = -f^0(x_0; -y)$ for each $y \in X$. This shows that f is strictly differentiable at x_0 and, hence, almost everywhere in U . $\square$

In order to establish that the essentially smooth functions are D -representable (see below) we shall need some further definitions.

A set-valued mapping Φ from a topological space U into subsets of a linear topological space (X, τ) is called a τ -*cusco* on U if

- (i) for each $x \in U$, $\Phi(x)$ is nonempty, convex, and compact;
- (ii) for each open subset W of X , $\{x \in U : \Phi(x) \subseteq W\}$ is open in U .

(Set-valued mappings satisfying only (ii) are called *upper or outer semicontinuous*.)

Our interest in cusco mappings arises from the fact that the Clarke subdifferential mapping of a locally Lipschitz function defined on a nonempty open subset of a Banach space is, a fortiori, a weak* cusco. Amongst the class of cusco mappings, special attention is given to the class of minimal cuscus. A cusco mapping Φ from a topological space U into subsets of a linear topological space (X, τ) is called a *minimal τ -cusco* if its graph does not strictly contain the graph of any other τ -cusco defined on U .

We may now present the following central theorem.

THEOREM 3.4. *Let U be a nonempty open subset of a Banach space X . Then the Clarke subdifferential mapping of each member of $\mathcal{S}_e(U)$ is a minimal weak* cusco on U .*

Proof. Let $f \in \mathcal{S}_e(U)$ and let us suppose, for the purpose of obtaining a contradiction, that $x \rightarrow \partial f(x)$ is not a minimal weak* cusco on U . That is, let us suppose that there exists a weak* cusco Φ on U whose graph is strictly contained in that of ∂f . It follows then, via a separation argument in (X^*, weak^*) , that we may find a point $x_0 \in U$ and a direction $\hat{y} \in \hat{X}$ (here $\hat{X}$ denotes the natural embedding of X into X^{**}) such that

$$\max \hat{y}(\Phi(x_0)) < \max \hat{y}(\partial f(x_0)).$$

Now consider the two cusco mappings $T_y : U \rightarrow 2^R$ and $S_y : U \rightarrow 2^R$ defined by

$$S_y(x) \equiv \hat{y}(\Phi(x)) \text{ and } T_y(x) \equiv \hat{y}(\partial f(x)).$$

Clearly, $S_y(x) \subseteq T_y(x)$ for each $x \in U$. However, $S_y(x_0) \neq T_y(x_0)$ since

$$\max S_y(x_0) = \max \hat{y}(\Phi(x_0)) < \max \hat{y}(\partial f(x_0)) = \max T_y(x_0).$$

So we will obtain our desired contradiction if we can show that $T_y = S_y$. Let $N \equiv \{x \in U : f^0(x; y) \neq -f^0(x; -y)\}$. It is clear then, from the definition of T_y , that

$$S_y(x) = T_y(x) = [-f^0(x; -y), f^0(x; y)] = \{f'(x; y)\} \text{ at each point of } U \setminus N.$$

We may now use the upper semicontinuity of S_y , a simple separation argument (in R), and Theorem 3.2 to deduce that $S_y = T_y$. $\square$

The importance of the previous result is that minimality of the Clarke subdifferential mapping is closely related to the, about to be defined, notion of D -representability. Moreover, minimality of the Clarke subdifferential mapping also provides a means for deducing differentiability results. Indeed, let us recall that a Banach space X is said to be of *class*($\mathcal{S}$) if every minimal weak* cusco that maps from a Baire space, into subsets of X^* , is single valued at the points of a dense and G_δ subset of its domain (see [10]). Consequently, we see that in these spaces the essentially smooth functions are strictly differentiable at the points of a dense and G_δ subset of their domain. The family of all *class*($\mathcal{S}$) spaces is rather large. Indeed, all *smooth* Banach spaces (that is, spaces which admit an equivalent norm that is Gâteaux differentiable everywhere except at 0) belong to this class (see [10]), as do all those Banach spaces that contain, as a dense subspace, the continuous linear image of an Asplund space (such spaces are called *GSG* spaces [10]).

Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a *class*($\mathcal{S}$) Banach space X . Then we say that f is *D-representable* on U if $D \equiv \{x \in U : \nabla f(x) \text{ exists}\}$ is dense in U and for *each* dense subset D^* of D ,

$$\partial f(x) = \overline{co}^{w^*} \{x^* \in X^* : x^* = \text{weak}^* - \lim_{\substack{x_n \rightarrow x \\ x_n \in D^*}} \nabla f(x_n)\}.$$

The next theorem now follows directly from Corollary 2.2 and Theorem 2.5 in [4].

THEOREM 3.5. *Let U be a nonempty open subset of a *class*($\mathcal{S}$) Banach space. Then each member of $\mathcal{S}_e(U)$ is D -representable.*

The next property that we shall address is that of integrability. Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a Banach space X . Then we say that f is *integrable* on U if $\partial(f - g) \equiv \{0\}$ for each real-valued locally Lipschitz function g , defined on U , with $\partial g(x) \subseteq \partial f(x)$ for each $x \in U$.

We now show that the essentially smooth functions are integrable.

PROPOSITION 3.6. *Suppose that U is a nonempty open subset of a Banach space X . Let f and g be real-valued locally Lipschitz functions on U such that $\partial g(x) \subseteq \partial f(x)$ for each $x \in U$. Then $\partial(f - g) \equiv \{0\}$ if and only if $x \rightarrow \partial(f - g)(x)$ is a minimal weak* cusco on U .*

Proof. Since $f = (f - g) + g$ we have that $\partial f(x) \subseteq \partial(f - g)(x) + \partial g(x) \subseteq \partial(f - g)(x) + \partial f(x)$ and “cancellation” of closed bounded convex sets implies that $0 \in \partial(f - g)(x)$ for all $x \in U$. So if $x \rightarrow \partial(f - g)(x)$ is a minimal weak* cusco on U , then $\partial(f - g) \equiv \{0\}$. The converse is obvious. $\square$

COROLLARY 3.7. *Suppose that U is a nonempty open connected subset of a Banach space X . Let f and g be real-valued locally Lipschitz functions defined on U such that $\partial g(x) \subseteq \partial f(x)$ for each $x \in U$. Then $f - g$ is constant on U if and only if $x \rightarrow \partial(f - g)(x)$ is a minimal weak* cusco on U .*

Proof. Suppose that $x \rightarrow \partial(f - g)(x)$ is a minimal weak* cusco on U . Let $x_0 \in U$, and let $U_0 \equiv \{x \in U : (f - g)(x) = (f - g)(x_0)\}$. It follows from the Mean-Value theorem and the result above that both U_0 and $U \setminus U_0$ are open subsets of U . Now, $U_0 \neq \emptyset$ since $x_0 \in U_0$. Therefore, $U_0 = U$, which shows that $f - g$ is constant on U . The converse is again obvious. $\square$

THEOREM 3.8. *Let U be a nonempty open subset of a Banach space X . Then each member of $\mathcal{S}_e(U)$ is integrable.*

Proof. Suppose that $f \in \mathcal{S}_e(U)$ and g is any real-valued locally Lipschitz function defined on U such that $\partial g(x) \subseteq \partial f(x)$ for all $x \in U$. Clearly then $g \in \mathcal{S}_e(U)$. Moreover, since $\partial(f - g)(x) \subseteq \partial f(x) - \partial g(x)$ we see that $f - g \in \mathcal{S}_e(U)$, and so the subdifferential mapping, $x \rightarrow \partial(f - g)(x)$, is a minimal weak* cusco on U . Hence, by the previous Proposition, $\partial(f - g) \equiv \{0\}$ on U . $\square$

We say that a locally Lipschitz function f is *hereditarily integrable* on an open subset U if its restriction to every open subset is integrable (on that open set). Hence, we see that the members of $\mathcal{S}_e(U)$ are not only integrable, but also hereditarily integrable on U . It is appropriate to recall that there are Lipschitz functions which are integrable but not hereditarily integrable (see [4, section 7]).

THEOREM 3.9 (identity theorem). *Suppose that f and g are real-valued locally Lipschitz functions defined on a nonempty open connected subset U of a Banach space X . If $f \in \mathcal{S}_e(U)$ and $x \rightarrow \partial g(x)$ is a minimal weak* cusco, then $f - g \equiv \text{constant}$ on U if and only if $\{x \in U : \partial g(x) \cap \partial f(x) \neq \emptyset\}$ is dense in U .*

Proof. Consider the set-valued mapping $T : U \rightarrow 2^{X^*}$ defined by $T(x) \equiv \partial g(x) \cap \partial f(x)$. Since both mappings, $x \rightarrow \partial f(x)$ and $x \rightarrow \partial g(x)$, are upper semicontinuous on U (and possess compact images) T possesses nonempty weak* compact convex images. Moreover, since the graphs of both ∂f and ∂g are closed in $U \times X^*$, with X^* equipped with the weak* topology, so is the graph of T . Hence, we may deduce that T is a cusco on U . However, for each $x \in U$, $T(x) \subseteq \partial f(x)$ and $T(x) \subseteq \partial g(x)$. Therefore, by the minimality of ∂f and ∂g we must have that $\partial g = T = \partial f$. The result now follows from Theorem 3.8. Yet again, the converse is obvious. $\square$

In the following theorem we provide a condition which is sufficient to ensure membership in $\mathcal{S}_e(U)$.

THEOREM 3.10. *Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a Banach space X . Let B be any subset of X such that $\overline{\text{sp}}B = X$. If for each $b \in B$, $\{x \in U : f^0(x; b) \neq -f^0(x; -b)\}$ is a Haar-null set, then $f \in \mathcal{S}_e(U)$.*

Proof. Let $y \in X \setminus \{0\}$. We will show that $\{x \in U : f^0(x; y) \neq -f^0(x; -y)\}$ is a Haar-null set. We may choose a countable set $B' \equiv \{b_n : n \in \mathbb{N}\} \subseteq B$ so that

$y \in \overline{sp}B'$. For each $n \in N$, let $S_n \equiv \{x \in U : f^0(x; b_n) = -f^0(x; -b_n)\}$. It is clear from the hypothesis that each set $U \setminus S_n$ is Haar-null. Let $S \equiv \bigcap \{S_n : n \in N\}$. We claim that f is strictly differentiable, in the direction y , at each point of S . So let us consider an arbitrary point $x_0 \in S$. To show that f is strictly differentiable at x_0 , in the direction y , we need only show that if x_1^* and x_2^* are Clarke subgradients of f at x_0 , then $x_1^*(y) = x_2^*(y)$ (this is because, if $f^0(x_0; y) \neq -f^0(x_0; -y)$, then we could construct, using the Hahn–Banach extension theorem, two distinct subgradients x_1^* and x_2^* of f , at x_0 , such that $x_1^*(y) = f^0(x_0; y)$ and $x_2^*(y) = -f^0(x_0; -y)$). To this end, let x_1^* and x_2^* be Clarke subgradients of f at x_0 . Then, $x_i^*(b_n) \leq f^0(x_0; b_n)$ and $x_i^*(-b_n) \leq f^0(x_0; -b_n)$ for each $i \in \{1, 2\}$ and $n \in N$. Therefore,

$$x_i^*(b_n) \leq f^0(x_0; b_n) = -f^0(x_0; -b_n) \leq -x_i^*(-b_n) = x_i^*(b_n)$$

for each $i \in \{1, 2\}$ and $n \in N$. Hence, $x_1^*(b_n) = f^0(x_0; b_n) = x_2^*(b_n)$ for each $n \in N$. Now, both x_1^* and x_2^* are linear on X , and so we have that $x_1^*(b) = x_2^*(b)$ for each $b \in spB'$. However, x_1^* and x_2^* are also both continuous on X ; therefore, $x_1^* = x_2^*$ on $\overline{sp}B'$. In particular, $x_1^*(y) = x_2^*(y)$. $\square$

Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a Banach space X . Then f is *upper hemismooth* (*lower hemismooth*) at a point $x \in U$, in the direction y , if

$$f^+(x; y) \geq \limsup_{t \rightarrow 0^+} f^0(x + ty; y) \quad \left(f^-(x; y) \leq \liminf_{t \rightarrow 0^+} -f^0(x + ty; -y) \right).$$

If f is both upper hemismooth and lower hemismooth at $x \in U$, in the direction y , then we say that f is *hemismooth* at x , in the direction y .

Remark 3.1. If we define

$$T_y : U \rightarrow R \text{ by } T_y(x) \equiv \limsup_{t \rightarrow 0^+} f^0(x + ty; y)$$

and

$$S_y : U \rightarrow R \text{ by } S_y(x) \equiv \liminf_{t \rightarrow 0^+} -f^0(x + ty; -y),$$

then it is easy to check that both T_y and S_y are Borel measurable on U . Hence, the set of points in U where f is upper (lower) hemismooth, in the direction y , is always a Borel subset of U . Indeed, to see that T_y is Borel measurable, it suffices to observe that

$$T_y(x) = \lim_{n \rightarrow \infty} g_n(x), \text{ where } g_n(x) \equiv \sup\{f^0(x + ty; y) : 0 < t \leq 1/n\},$$

and

$$g_n(x) \equiv \lim_{m \rightarrow \infty} f_m^n(x), \text{ where } f_m^n(x) \equiv \max\{f^0(x + ty; y) : 1/m \leq t \leq 1/n\}$$

(for each $m > n$), are upper semicontinuous on U . A similar argument shows that S_y is also Borel measurable.

Further, we say that f is *essentially upper hemismooth* (*essentially lower hemismooth*) on U , in the direction y , if the set of all points in U where f is **not** upper hemismooth (lower hemismooth), in the direction y , is a Haar-null set. We shall also say that f is *pseudoregular* at x , in the direction y , if $f^0(x; y) = f^+(x; y)$, and we shall say that f is pseudoregular at x if it is pseudoregular at x in every direction y .

LEMMA 3.11. *Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a Banach space X . Then, in the notation of the previous remark, for each $y \in S(X)$ the (Borel) set*

$$F_y \equiv \{x \in U : f^0(x; y) > T_y(x)\} \quad (E_y \equiv \{x \in U : -f^0(x; -y) < S_y(x)\})$$

has the property that for each $x_0 \in U$, $F_y(x_0) \equiv \{r \in R : x_0 + ry \in F_y\}$ ($E_y(x_0) \equiv \{r \in R : x_0 + ry \in E_y\}$) is at most countable.

Proof. Fix $y \in S(X)$ and $x_0 \in U$. We will show that $F_y(x_0)$ is at most countable (the proof that $E_y(x_0)$ is countable is identical to this). Note that without loss of generality we may assume that $F_y(x_0)$ is nonempty. So in this case, we define $s : F_y(x_0) \rightarrow \mathbf{Q}^2$ by $s(t) \equiv (r_1, r_2)$, where $r_1 \in (T_y(x_0 + ty), f^0(x_0 + ty; y)) \cap \mathbf{Q}$ and $r_2 \in (t, \infty) \cap \mathbf{Q}$ is chosen so that

$$\sup\{f^0(x_0 + ry; y) : t < r < r_2\} < r_1.$$

It is easy to see that s is 1 to 1, and so F_y must be at most countable. Indeed, if $t_1 < t_2$ and $s(t_1) = s(t_2) = (r_1, r_2)$, then $t_1 < t_2 < r_2$ and so $f^0(x_0 + t_2y; y) < r_1$. However, this is impossible since it contradicts the definition of s at t_2 . (Here, $\mathbf{Q}$ denotes the rational numbers.) $\square$

Remark 3.2. If X is a Banach space, then for each $y \in S(X)$, $\{x \in U : f \text{ is upper hemismooth, but not pseudoregular, in the direction } y\}$ is contained in F_y and hence is a Haar-null set (in this case, we may take the normalized Lebesgue measure, supported on $sp\{y\}$, as a test measure for the Borel set F_y).

LEMMA 3.12. *Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a Banach space X . If for some $y \in X \setminus \{0\}$ f is essentially upper hemismooth (essentially lower hemismooth) in the direction y , then $f^0(x; y) = f'(x; y)$ ($-f^0(x; -y) = f'(x; y)$) almost everywhere in U .*

Proof. Suppose that f is essentially upper hemismooth in the direction y (the proof for the case when f is essentially lower hemismooth in the direction y is similar). It follows from Remark 3.2 that $f^0(x; y) = f^+(x; y)$ almost everywhere in U . It now only remains to observe that $f^+(x; y) = f'(x; y)$ almost everywhere in U (to see this, take the normalized Lebesgue measure, supported on $sp\{y\}$, as a test measure for the set $\{x \in U : f'(x; y) \text{ does not exist}\}$). $\square$

Now from Lemma 3.12 and Theorem 3.10 we may deduce the following theorem.

THEOREM 3.13. *Let f be a real-valued locally Lipschitz function defined on a nonempty open subset U of a Banach space X . Let B be any subset of X such that $\overline{sp}B = X$. If for each $b \in B$ either, (i) f is essentially upper hemismooth in the directions b and $-b$; (ii) f is essentially lower hemismooth in the directions b and $-b$; (iii) f is essentially hemismooth in the direction b ; or, (iv) f is essentially hemismooth in the direction $-b$, then $f \in \mathcal{S}_e(U)$.*

It follows from the above theorem that the class of essentially smooth functions is quite large, and as we shall see next, also quite robust. More precisely, we will show that the essentially smooth functions are closed under addition, subtraction, multiplication, and division (when this is defined), as well as the lattice operations. Actually, it not unreasonable to expect that the essentially smooth functions obey closure properties considerably stronger than those just mentioned. In fact, a first, but naive, guess might be that if $f_1, f_2, \dots, f_n \in \mathcal{S}_e(U)$ and $g \in \mathcal{S}_e(R^n)$, then $g \circ f \in \mathcal{S}_e(U)$, where $f \equiv (f_1, f_2, \dots, f_n)$. To the contrary, in [5], the authors give an example that shows that this is not true when $n \geq 2$.

For the final result of the paper we will need to consider vector-valued functions. Let U be a nonempty open subset of a Banach space X , and let V be a nonempty open subset of R^n . If $x : U \rightarrow V$ is defined by

$$x(t) \equiv (x_1(t), x_2(t), \dots, x_n(t)) \quad \text{with } x_j : U \rightarrow R,$$

then we say that the vector-valued function x is *essentially smooth* on U if $x_j \in \mathcal{S}_e(U)$ for each $1 \leq j \leq n$ and in this case we write $x \in \mathcal{S}_e(U; V)$. In addition to this, we will say that a real-valued locally Lipschitz function f defined on a nonempty open subset V on R^n is *arc-wise essentially smooth* on V , if for each locally Lipschitz function $x \in \mathcal{S}_e((0, 1); V)$

$$\lambda(\{t \in (0, 1) : f^0(x(t); x'(t)) \neq -f^0(x(t); -x'(t))\}) = 0.$$

$$[\text{Here } x'(t) \equiv (x'_1(t), x'_2(t), \dots, x'_n(t)).]$$

We shall denote by $\mathcal{A}_e(V)$ the family of all arc-wise essentially smooth Lipschitz functions on V . A relevant fact for the next theorem is that $\mathcal{A}_e(V)$ contains all the pseudoregular and semismooth (see [18] for definition) functions on V . In particular, $\mathcal{A}_e(V)$ contains all the continuous convex and C^1 functions on V (see [5]).

THEOREM 3.14 (see [5, Theorem 2.3]). *Let U be a nonempty open subset of a Banach space X , and let $f \equiv (f_1, f_2, \dots, f_n) \in \mathcal{S}_e(U; V)$, where V is any nonempty open subset of R^n that contains $f(U)$. Then $g \circ f \in \mathcal{S}_e(U)$ whenever $g \in \mathcal{A}_e(V)$.*

Remark 3.3. In [4] it is shown that if U is a nonempty open subset of R , then $\mathcal{S}_e(U) = \mathcal{A}_e(U)$. However, it is also shown in [4] that if U is a nonempty open subset of R^n ($n \geq 2$), then $\mathcal{A}_e(U)$ is a proper subset of $\mathcal{S}_e(U)$.

REFERENCES

- [1] C. BERG, J. P. R. CHRISTENSEN, AND P. RESSEL, *Harmonic Analysis on Semigroups*, Springer-Verlag, New York, 1984.
- [2] J. M. BORWEIN, *Minimal cuscus and subgradients of Lipschitz functions*, in Fixed Point Theory and its Applications, J.-B. Baillon and M. Thera, eds., Pitman Lecture Notes in Math., Longman Press, London, 1991, pp. 57–82.
- [3] J. M. BORWEIN AND S. P. FITZPATRICK, *Closed convex Haar null sets*, CECM report 95:052, Simon Fraser University, Burnaby, BC, Canada.
- [4] J. M. BORWEIN AND W. B. MOORS, *Essentially smooth Lipschitz functions*, J. Funct. Anal., 149 (1997), pp. 305–351.
- [5] J. M. BORWEIN AND W. B. MOORS, *A chain rule for essentially smooth Lipschitz functions*, SIAM J. Optim., 8 (1998), pp. 300–307.
- [6] J. M. BORWEIN AND W. B. MOORS, *Lipschitz functions with minimal Clarke subdifferential mappings*, in Proc. Optim. Miniconference III: The University of Melbourne, July 18, 1996, B. M. Glover, B. D. Craven, and D. Ralph, eds., Ballarat, Victoria, 1997, pp. 5–12.
- [7] J. P. R. CHRISTENSEN, *On sets of Haar measure zero in Abelian groups*, Israel J. Math., 13 (1972), pp. 255–260.
- [8] F. H. CLARKE, *Optimization and Nonsmooth Analysis*, John Wiley, New York, 1983.
- [9] R. M. DUDLEY, *Real Analysis and Probability*, Wadsworth and Brooks/Cole Math. Ser., Wadsworth and Brooks/Cole Software, Pacific Grove, CA, 1989.
- [10] M. FABIAN, *Gateaux Differentiability of Convex Functions and Topology—Weak Asplund Spaces*, John Wiley and Sons, Inc., Adv. Books, New York, 1997.
- [11] B. R. HUNT, T. SAUER, AND J. A. YORKE, *Prevalence: A translation-invariant “almost every” on infinite-dimensional spaces*, Bull. Amer. Math. Soc., 27 (1992), pp. 217–238.
- [12] J. E. JAYNE, I. NAMIOKA, AND C. A. ROGERS, *Topological properties of Banach spaces*, Proc. London Math. Soc. (3), 66 (1993), pp. 651–672.
- [13] A. S. KECHRIS, *Topology and descriptive set theory*, Topology Appl., 58 (1994), pp. 195–222.

- [14] V. L. KLEE, *Invariant metrics in groups (Solution of a problem of Banach)*, Proc. Amer. Math. Soc., 3 (1952), pp. 484–487.
- [15] P. MANKIEWICZ, *On the differentiability of Lipschitz mappings in Frechet spaces*, Studia Math., 45 (1973), pp. 15–29.
- [16] E. MATOUSKOVA, *Convexity and Haar null sets*, Proc. Amer. Math. Soc., 125 (1997), pp. 1793–1799.
- [17] E. MATOUSKOVA AND C. STEGALL, *A characterization of reflexive Banach spaces*, Proc. Amer. Math. Soc., 144 (1996), pp. 1083–1090.
- [18] R. MIFFLIN, *Semismooth and semiconvex functions in optimization*, SIAM J. Control Optim., 15 (1977), pp. 959–972.
- [19] W. B. MOORS, *A characterisation of minimal subdifferential mappings of locally Lipschitz functions*, Set-Valued Anal., 3 (1995), pp. 129–141.
- [20] R. R. PHELPS, *Gaussian null sets and differentiability of Lipschitz functions on Banach spaces*, Pacific J. Math., 77 (1978), pp. 523–531.
- [21] E. SZPILRAJN, *Sur certain invariants de l'operation $(\mathcal{A})$* , Fund. Math., 21 (1933), pp. 229–235.

PRIMAL-DUAL INTERIOR-POINT METHODS FOR SELF-SCALED CONES*

YU. E. NESTEROV[†] AND M. J. TODD[‡]

Abstract. In this paper we continue the development of a theoretical foundation for efficient primal-dual interior-point algorithms for convex programming problems expressed in conic form, when the cone and its associated barrier are *self-scaled* (see Yu. E. Nesterov and M.J. Todd, *Math. Oper. Res.*, 22 (1997), pp. 1–42). The class of problems under consideration includes linear programming, semidefinite programming, and convex quadratically constrained, quadratic programming problems. For such problems we introduce a new definition of affine-scaling and centering directions. We present efficiency estimates for several symmetric primal-dual methods that can loosely be classified as path-following methods. Because of the special properties of these cones and barriers, two of our algorithms can take steps that typically go a large fraction of the way to the boundary of the feasible region, rather than being confined to a ball of unit radius in the local norm defined by the Hessian of the barrier.

Key words. convex programming, conical form, interior-point algorithms, self-concordant barrier, self-scaled cone, self-scaled barrier, path-following algorithms, symmetric primal-dual algorithms

AMS subject classifications. 90C05, 90C25, 65Y20

PII. S1052623495290209

1. Introduction. This paper continues the development begun in [NT97] of a theoretical foundation for efficient interior-point methods for problems that are extensions of linear programming. Here we concentrate on symmetric primal-dual algorithms that can loosely be classified as path-following methods. While standard form linear programming problems minimize a linear function of a vector of variables subject to linear equality constraints and the requirement that the vector belong to the nonnegative orthant in n -dimensional Euclidean space, here this cone is replaced by a possibly nonpolyhedral convex cone. Note that any convex programming problem can be expressed in this conical form.

Nesterov and Nemirovskii [NN94] have investigated the essential ingredients necessary to extend several classes of interior-point algorithms for linear programming (inspired by Karmarkar's famous projective-scaling method [Ka84]) to nonlinear settings. The key element is that of a *self-concordant barrier* for the convex feasible region. This is a smooth convex function defined on the interior of the set, tending to $+\infty$ as the boundary is approached, that together with its derivatives satisfies certain Lipschitz continuity properties. The barrier enters directly into functions used in path-following and potential-reduction methods, but, perhaps as importantly, its Hessian at any point defines a local norm whose unit ball, centered at that point, lies

*Received by the editors August 11, 1995; accepted for publication (in revised form) February 4, 1997.

<http://www.siam.org/journals/siopt/8-2/29020.html>

[†]CORE, Catholic University of Louvain, Louvain-la-Neuve 1348, Belgium (nesterov@core.ucl.ac.be). This research was partially supported by the Russian Fund of Fundamental Studies, grant 96-01-00293.

[‡]School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY 14853 (miketodd@cs.cornell.edu). Some of this work was done while the author was on a sabbatical leave from Cornell University visiting the Department of Mathematics at the University of Washington and the Operations Research Center at the Massachusetts Institute of Technology. This research was supported in part by the University of Washington and NSF grants CCR-9103804, DMS-9303772, and DDM-9158118.

completely within the feasible region. Moreover, the Hessian varies in a well-controlled way in the interior of this ball.

In [NT97], Nesterov and Todd introduced a special class of *self-scaled* convex cones and associated barriers. While they are required to satisfy certain apparently restrictive conditions, this class includes some important instances, for example, the cone of (symmetric) positive semidefinite matrices and the second-order cone, as well as the nonnegative orthant in R^n . In fact, these cones (and their direct products) are perhaps the only self-scaled cones interesting for optimization. It turns out that self-scaled cones coincide with homogeneous self-dual cones as was pointed out by Güler [Gu96] (see also the discussion in [NT97]), and these have been characterized as being built up via direct products from the cones listed above, cones of positive semidefinite Hermitian complex and quaternion matrices, and one exceptional cone. However, for the reasons given in detail at the end of this introduction, we believe that it is simpler to carry out the analysis and develop algorithms in the abstract setting of self-scaled cones. We maintain the name self-scaled to emphasize our primary interest in the associated self-scaled barriers.

For such cones, the Hessian of the barrier at any interior point maps the cone onto its dual cone, and vice versa for the conjugate barrier. In addition, for any pair of points, one in the interior of the original (primal) cone and the other in the interior of the dual cone, there is a unique *scaling* point at which the Hessian carries the first into the second. Thus there is a very rich class of scaling transformations, which come from the Hessians evaluated at the points of the cone itself (hence, self-scaled).

These conditions have extensive consequences. For our purposes, the key results are the existence of a symmetric primal-dual scaling and the fact that good approximations of self-scaled barriers and their gradients extend far beyond unit balls defined by the local norm, and are in fact valid up to a constant fraction of the distance to the boundary in any direction. Using these ideas [NT97] developed primal long-step potential-reduction and path-following algorithms as well as a symmetric long-step primal-dual potential-reduction method.

In this paper we present some new properties of self-scaled cones and barriers that are necessary for deriving and analyzing primal-dual path-following interior-point methods.

The first part of the paper continues the study of the properties of self-scaled barriers started in [NT97]. In section 2 we introduce the problem formulation and state the main definitions and our notation. In section 3 we prove some symmetry results for primal and dual norms and study the properties of the third derivative of a self-scaled barrier and the properties of the scaling point. In section 4 we introduce the definitions of the primal-dual central path and study different proximity measures. Section 5 is devoted to generalized affine-scaling and centering directions.

The second part of the paper applies these results to the derivation of short- and long-step symmetric primal-dual methods that follow the central path in some sense, using some proximity measure. Primal-dual path-following methods like those of Monteiro and Adler [MA89] and Kojima, Mizuno, and Yoshise [KMY89] are studied in section 6. These methods follow the path closely using a local proximity measure and take short steps. At the end of this section we consider an extension of the predictor-corrector method of Mizuno, Todd, and Ye [MTY90], which uses the affine-scaling and centering directions of section 5, also relying on the same proximity measure. This algorithm employs an adaptive step size rule. In section 7 we present another predictor-corrector algorithm, a variant of the functional-proximity path-following

scheme proposed by Nesterov [Ne96] for general nonlinear problems. In the case of self-scaled cones the method is based on the affine-scaling and centering directions. This algorithm uses a global proximity measure and allows considerable deviation from the central path; thus much longer steps are possible than in the algorithms of the previous section. Nevertheless, the results here can also be applied to the predictor-corrector method considered there.

All these methods require $O(\sqrt{\nu} \ln(1/\epsilon))$ iterations to generate a feasible solution with objective function within ϵ of the optimal value, where ν is a parameter of the cone and barrier corresponding to n for the nonnegative orthant in R^n . All are variants of methods already known for the standard linear programming case or for the more general conic case, but we stress the improvements possible because the cone is assumed self-scaled. For example, for the predictor-corrector methods of the last two sections we can prove that the predictor step size is at least a constant times the square root of the maximal possible step size along the affine-scaling direction. Alternatively, it can be bounded from below by a constant times the maximal possible step size divided by $\nu^{1/4}$. These results were previously unknown even in the context of linear programming.

In order to motivate our development, let us explain why we carry out our analysis in the abstract setting of self-scaled cones, rather than restricting ourselves to say semidefinite programming. After all, the latter context includes linear programming, and, as has been shown by several authors including Nesterov and Nemirovskii [NN94], convex quadratically constrained quadratic programming. However, in the latter case, the associated barrier parameter from the semidefinite programming formulation is far larger than it need be, and the complexity of the resulting algorithm also correspondingly too large. In contrast, using the second-order (or Lorentz, or ice cream) cone permits one to employ the appropriate barrier and corresponding efficient algorithms. The abstract setting of self-scaled cones and barriers allows us to treat each problem with its natural formulation and barrier. Thus if a semidefinite programming problem has a positive semidefinite matrix as a variable and also involves inequality constraints, we do not need to use the slack variables for the constraints to augment the positive semidefinite matrix and thus deal with block diagonal matrices; instead, our variable is just the combination of the original matrix variable and the slack vector, and the self-scaled cone in which it is constrained to lie is the product of a positive semidefinite cone and a nonnegative orthant of appropriate dimensions. In this way, it is in fact easier, although more abstract, to work in the more general setting. Similarly, nonnegative variables are dealt with directly (with the cone the nonnegative orthant), rather than embedded into the diagonal of a positive semidefinite matrix, so our algorithms apply naturally to linear programming as well as semidefinite programming, without having to consider what happens when the appropriate matrices are diagonal. We can also handle more general constraints, such as convex quadratic inequalities on symmetric positive semidefinite matrices, by using combinations of the positive semidefinite cone and the second-order cone. Finally, we can write linear transformations in the general form $x \rightarrow Ax$, rather than considering specific forms for semidefinite programming such as $X \rightarrow PXP^T$; similar comments apply to the bilinear and trilinear expressions that arise when dealing with second and third derivatives. Thus we obtain unified methods and unified analyses for methods that apply to all of these cases directly, rather than through reformulations; moreover, we see exactly what properties of the cones and barriers lead to the desired results and algorithms. We will explain several of our constructions for the case of linear or

semidefinite programming to aid the reader, but the development will be carried out in the framework of abstract self-scaled cones for the reasons previously given.

In what follows we often refer to different statements of [NN94] and [NT97]. The corresponding references are indicated by asterisks. An upper case asterisk in the reference $T^* (C^*, D^*, P^*)$ 1.1.1 corresponds to the first theorem (corollary, definition, proposition) in the first section of Chapter 1 of [NN94]. A lower case asterisk, for example, T_* 1.1 (or $(1.1)_*$), corresponds to the first theorem (or equation) in section 1 of [NT97].

2. Problem formulation and notation. Let K be a closed convex cone in a finite-dimensional real vector space E (of dimension at least 1) with dual space E^* . We denote the corresponding scalar product by $\langle s, x \rangle$ for $x \in E$, $s \in E^*$. In what follows we assume that the interior of the cone K is nonempty and that K is *pointed* (contains no straight line).

The problem we are concerned with is

$$(2.1) \quad \begin{array}{ll} \text{(P)} & \min \quad \langle c, x \rangle \\ & \text{s.t.} \quad Ax = b, \\ & \quad \quad x \in K, \end{array}$$

where we assume that

$$(2.2) \quad \begin{array}{l} A \text{ is a surjective linear operator from } E \text{ to} \\ \text{another finite-dimensional real vector space } Y^*. \end{array}$$

Here, $b \in Y^*$ and $c \in E^*$. The assumption that A is surjective is without loss of generality (or else replace Y^* with its range).

Define the cone K^* dual to K as follows:

$$K^* := \{s \in E^* : \langle s, x \rangle \geq 0, \forall x \in K\}.$$

Note that K^* is also a pointed cone with a nonempty interior. Then the dual to problem (P) is (see [ET76])

$$(2.3) \quad \begin{array}{ll} \text{(D)} & \max \quad \langle b, y \rangle \\ & \text{s.t.} \quad A^*y + s = c, \\ & \quad \quad s \in K^*, \end{array}$$

where A^* denotes the adjoint of A , mapping Y to E^* , and $y \in Y$.

If K is the nonnegative orthant in R^n , then (P) and (D) are the standard form linear programming problem and its dual; if K is the cone of positive semidefinite matrices of order n , $\langle c, x \rangle := \text{Trace}(c^T x)$, and $Ax := (\langle a_k, x \rangle) \in R^m$ where the a_k 's are symmetric matrices of order n , we obtain the standard form semidefinite programming problem and its dual; here A^*y is $\sum y_k a_k$. In both these cases, K^* can be identified with K . The surjectivity assumption requires that the constraint matrix have full row rank in the first case, and that the a_k 's be linearly independent in the second.

We make the following assumptions about (P) and (D):

$$(2.4) \quad S^0(P) := \{x \in \text{int } K : Ax = b\} \text{ is nonempty,}$$

and

$$(2.5) \quad S^0(D) := \{(y, s) \in Y \times \text{int } K^* : A^*y + s = c\} \text{ is nonempty.}$$

These assumptions imply (see [NN94], T^* 4.2.1) that both (P) and (D) have optimal solutions, their optimal values are equal, and the sets of optimal solutions of both problems are bounded (see [Ne96]). Also, it is easy to see that, if x and (y, s) are feasible in (P) and (D) respectively, then

$$\langle c, x \rangle - \langle b, y \rangle = \langle s, x \rangle.$$

This quantity is the (nonnegative) *duality gap*.

Our algorithms for (P) and (D) require that the cone K be *self-scaled*, i.e., admit a *self-scaled barrier*. We now introduce the terminology to define this concept.

Let F be a ν -self-concordant logarithmically homogeneous barrier for cone K (see D^* 2.3.2). Recall that by definition, F is a self-concordant barrier for K (see D^* 2.3.1), which for all $x \in \text{int } K$ and $\tau > 0$ satisfies the identity

$$(2.6) \quad F(\tau x) \equiv F(x) - \nu \ln \tau.$$

Since K is a pointed cone, $\nu \geq 1$ in view of C^* 2.3.3. The reader might want to keep in mind for the properties below the trivial example where K is the nonnegative half-line and F the function $-\ln(\cdot)$, with $\nu = 1$. The two main interesting cases are where K is the nonnegative orthant R_+^n and F the standard logarithmic barrier function ($F(x) := -\ln(x) := -\sum_{j=1}^n \ln x^{(j)}$), and where K is the cone S_+^n of symmetric positive semidefinite $n \times n$ matrices and F the log-determinant barrier function ($F(x) := -\ln \det x$); in both cases, $\nu = n$. In the first case, $F'(x) = -([x^{(j)}]^{-1})$, the negative of the vector of reciprocals of the components of x , with $F''(x) = X^{-2}$, where X is the diagonal matrix containing the components of x ; in the second, $F'(x) = -x^{-1}$, the negative of the inverse matrix, while $F''(x)$ is the linear transformation defined by $F''(x)v = x^{-1}vx^{-1}$.

We will often use the following straightforward consequences of (2.6): for all $x \in \text{int } K$ and $\tau > 0$,

$$(2.7) \quad F'(\tau x) = \frac{1}{\tau} F'(x), \quad F''(\tau x) = \frac{1}{\tau^2} F''(x),$$

$$(2.8) \quad F''(x)x = -F'(x), \quad F'''(x)[x] = -2F''(x),$$

$$(2.9) \quad \langle F'(x), x \rangle = -\nu,$$

$$(2.10) \quad \langle F''(x)x, x \rangle = \nu, \quad \langle F'(x), [F''(x)]^{-1}F'(x) \rangle = \nu$$

(see P^* 2.3.4).

Let the function F_* on $\text{int } K^*$ be conjugate to F , namely,

$$(2.11) \quad F_*(s) := \max\{-\langle s, x \rangle - F(x) : x \in K\}.$$

In accordance with T^* 2.4.4, F_* is a ν -self-concordant logarithmically homogeneous barrier for K^* . (If $F(x) = -\ln(x)$ for $x \in \text{int } R_+^n$, $F_*(s) = -\ln(s) - n$, while if $F(x) = -\ln \det x$ for $x \in \text{int } S_+^n$, $F_*(s) = -\ln \det s - n$.) We will often use the following properties of conjugate self-concordant barriers for dual cones: for any $x \in \text{int } K$ and $s \in \text{int } K^*$,

$$(2.12) \quad -F'(x) \in \text{int } K^*, \quad -F'_*(s) \in \text{int } K,$$

$$(2.13) \quad F_*(-F'(x)) = \langle F'(x), x \rangle - F(x) = -\nu - F(x)$$

(using (2.9)),

$$(2.14) \quad F(-F'_*(s)) = -\nu - F_*(s),$$

$$(2.15) \quad F'_*(-F'(x)) = -x, \quad F'(-F'_*(s)) = -s,$$

$$(2.16) \quad F''(-F'_*(s)) = [F''_*(s)]^{-1}, \quad F''_*(-F'(x)) = [F''(x)]^{-1},$$

$$(2.17) \quad F(x) + F_*(s) \geq -\nu + \nu \ln \nu - \nu \ln \langle s, x \rangle,$$

and the last inequality is satisfied as an equality if and only if $s = -\alpha F'(x)$ for some $\alpha > 0$ (see P^* 2.4.1).

In this paper, as in [NT97], we consider cones and barriers of a rather special type. Let us give our main definition.

DEFINITION 2.1. *Let K be a pointed cone with nonempty interior and let F be a ν -self-concordant logarithmically homogeneous barrier for cone K . We call F a ν -self-scaled barrier for K if for any w and x from $\text{int } K$,*

$$(2.18) \quad F''(w)x \in \text{int } K^*$$

and

$$(2.19) \quad F_*(F''(w)x) = F(x) - 2F(w) - \nu.$$

If K admits such a barrier, we call it a self-scaled cone.

Note that the identity (2.19) has an equivalent form: if $x \in \text{int } K$ and $s \in \text{int } K^*$ then

$$(2.20) \quad F([F''(x)]^{-1}s) = F_*(s) + 2F(x) + \nu$$

(merely write x for w and $[F''(x)]^{-1}s$ for x in (2.19)). Tunçel [Tu96] has given a simpler equivalent form (using (2.15)): if $F''(w)x = -F'(v)$, then

$$F(w) = \frac{[F(x) + F(v)]}{2}.$$

In fact, self-scaled cones coincide with homogeneous self-dual cones (see Güler [Gu96] and the discussion in [NT97]), but we will maintain the term self-scaled to emphasize our interest in the associated self-scaled barriers.

One important property of a self-scaled barrier is the existence of scaling points (see T_* 3.2): for every $x \in \text{int } K$ and $s \in \text{int } K^*$, there is a unique point $w \in \text{int } K$ satisfying

$$F''(w)x = s.$$

For example, when K is the nonnegative orthant in R^n the scaling point w is given by the following componentwise expression:

$$w^{(j)} = \left[\frac{x^{(j)}}{s^{(j)}} \right]^{1/2};$$

when K is the cone of positive semidefinite matrices of order n , then

$$w = x^{1/2}[x^{1/2}sx^{1/2}]^{-1/2}x^{1/2} = s^{-1/2}(s^{1/2}xs^{1/2})^{1/2}s^{-1/2}.$$

In what follows we always assume that cone K is self-scaled and F is a corresponding ν -self-scaled barrier. Even though we know that we are in a self-dual setting, so that we can identify E and E^* and K and K^* , we prefer to keep the distinction, which allows us to distinguish primal and dual norms merely by their arguments, and makes sure that our methods are scale-invariant; indeed, it is impossible to write methods that are not, such as updating x to $x - F'(x)$ or to $x - s$, since the points lie in different spaces. We note that the two examples given above, the logarithmic barrier function for the nonnegative orthant in R^n and the log-determinant barrier for the cone of symmetric positive semidefinite $n \times n$ matrices, are both self-scaled barriers with parameter n . For example, in the second case we find $F''(v)x = v^{-1}xv^{-1}$ is positive definite if x and v are, with

$$F_*(F''(v)x) = -\ln \det(v^{-1}xv^{-1}) - n = -\ln \det x + 2 \ln \det v - n = F(x) - 2F(v) - n,$$

as desired. In addition, [NT97] shows that the standard barrier for the second-order cone is self-scaled, and also that the Cartesian product of self-scaled cones is also self-scaled, with the associated barrier the sum of the individual self-scaled barriers on the components; its parameter is just the sum of the parameters of the individual barriers.

3. Some results for self-scaled barriers. In this section we present some properties of self-scaled barriers, which complement the results of sections 3_{*}–5_{*}. The reader may prefer to skip the proofs at first reading.

We will use the following convention in notation: the meaning of the terms $\|u\|_v$, $|u|_v$ and $\sigma_v(u)$ is completely defined by the spaces of arguments (which are always indicated explicitly in the text). Thus, if $v \in \text{int } K$ then

- $\|u\|_v$ means $\langle F''(v)u, u \rangle^{1/2}$ if $u \in E$, and $\langle u, [F''(v)]^{-1}u \rangle^{1/2}$ if $u \in E^*$;
- $\sigma_v(u) := 1/\alpha$, where $\alpha > 0$ is either the maximal possible step (possibly $+\infty$) in the cone K such that $v - \alpha u \in K$, if $u \in E$, or the maximal possible step in the cone K^* such that $-F'(v) - \alpha u \in K^*$, if $u \in E^*$; equivalently, $\sigma_v(u)$ is the minimum nonnegative β such that either $\beta v - u \in K$, if $u \in E$, or $-\beta F'(v) - u \in K^*$, if $u \in E^*$.

Similarly, if $v \in \text{int } K^*$ then

- $\|u\|_v$ means $\langle u, F''_*(v)u \rangle^{1/2}$ if $u \in E^*$, and $\langle [F''_*(v)]^{-1}u, u \rangle^{1/2}$ if $u \in E$;
- $\sigma_v(u) := 1/\alpha$, where $\alpha > 0$ is either the maximal possible step in the cone K^* such that $v - \alpha u \in K^*$, if $u \in E^*$, or the maximal possible step in the cone K such that $-F'_*(v) - \alpha u \in K$, if $u \in E$; equivalently, $\sigma_v(u)$ is the minimum nonnegative β such that either $\beta v - u \in K^*$, if $u \in E^*$, or $-\beta F'_*(v) - u \in K$, if $u \in E$.

In this notation

$$|u|_v := \max\{\sigma_v(u), \sigma_v(-u)\}.$$

For a self-adjoint linear operator B from E to E^* or vice versa we define its norm as follows: for $v \in \text{int } K$ or $v \in \text{int } K^*$, let

$$\|B\|_v := \max\{\|Bu\|_v : \|u\|_v \leq 1\},$$

where u belongs to the domain of B .

Thus if, for example, $x \in \text{int } K$, $v, z \in E$, and $B : E \rightarrow E^*$, then

$$(3.1) \quad \langle Bv, z \rangle \leq \|Bv\|_x \cdot \|z\|_x \leq \|B\|_x \cdot \|v\|_x \cdot \|z\|_x.$$

Also note that

$$(3.2) \quad \|B\|_x \leq \alpha \Leftrightarrow -\alpha F''(x) \leq B \leq \alpha F''(x).$$

Let us illustrate these definitions when E and E^* are R^n , and K and K^* their nonnegative orthants. Let $u.*v$ ($u./v$) denote the vector whose components are the products (ratios) of the corresponding components of u and v . Then $\|u\|_v$ is the 2-norm of $u./v$ if u and v belong to the same space and of $u.*v$ if they belong to different spaces; $\sigma_v(u)$ is the maximum of 0 and the maximum component of $u./v$ or of $u.*v$; and $|u|_v$ is the infinity- or max-norm of the appropriate vector. Suppose next that E and E^* are spaces of symmetric matrices, and K and K^* their cones of positive semidefinite elements. Writing $\chi(x)$ for the vector of eigenvalues of the matrix x and $\chi_{\max}(x)$ and $\chi_{\min}(x)$ for its largest and smallest eigenvalues, we find that $\|u\|_v$ is the 2-norm of $\chi(v^{-1/2}uv^{-1/2})$ if both u and v lie in the same space, or of $\chi(v^{1/2}uv^{1/2})$, otherwise. Similarly, $\sigma_v(u)$ is the maximum of 0 and $\chi_{\max}(v^{-1/2}uv^{-1/2})$ or $\chi_{\max}(v^{1/2}uv^{1/2})$, and $|u|_v$ is the infinity- or max-norm of the appropriate vector of eigenvalues. Finally, if $B : E \rightarrow E^*$ is defined by $Bu := w^{-1}uw^{-1}$, with u and w in E , then for $v \in E$, $\|B\|_v = \max\{\|\chi(v^{1/2}w^{-1}uw^{-1}v^{1/2})\|_2 : \|\chi(v^{-1/2}uv^{-1/2})\|_2 \leq 1\} = \max\{\|\chi(v^{1/2}w^{-1}v^{1/2}xv^{1/2}w^{-1}v^{1/2})\|_2 : \|\chi(x)\|_2 \leq 1\}$. It can be seen that our general notation is somewhat less cumbersome.

Returning to the general case, let us first demonstrate in the following that $|\cdot|_v$ is indeed a norm (it is clear that $\|\cdot\|_v$ is).

PROPOSITION 3.1. $|\cdot|_v$ defines a norm on E and on E^* , for v in $\text{int } K$ or $\text{int } K^*$.

Proof. Without loss of generality, we assume $v \in \text{int } K$ and that we are considering $|\cdot|_v$ defined on E . From the definition it is clear that each of $\sigma_v(z)$ and $\sigma_v(-z)$ is positively homogeneous, and hence so is $|\cdot|_v$. Moreover, clearly $|-z|_v = |z|_v$. It is also immediate that $\sigma_v(z) = 0$ if $-z \in K$, and (using the second definition of $\sigma_v(z)$) that the converse holds also. Thus, since K is pointed, $|z|_v = 0$ iff $z = 0$. It remains to show that $|\cdot|_v$ is subadditive.

Let $\beta_i := \sigma_v(z_i)$ for $i = 1, 2$. Then $\beta_i v - z_i \in K$ for each i , so that $(\beta_1 + \beta_2)v - (z_1 + z_2) \in K$, and thus $\sigma_v(z_1 + z_2) \leq \beta_1 + \beta_2$. We conclude that $\sigma_v(z_1 + z_2) \leq \sigma_v(z_1) + \sigma_v(z_2) \leq |z_1|_v + |z_2|_v$. This inequality also holds when we change the signs of the z_i 's, and then taking the maximum shows that $|\cdot|_v$ is subadditive as required. $\square$

Henceforth, we usually follow the convention that s , t , and u belong to E^* , with s and t in $\text{int } K^*$, while p , v , w , x , and z belong to E , with x and w in $\text{int } K$. We reserve w for the scaling point such that $F''(w)x = s$.

3.1. Some symmetry results. Let us prove several symmetry results for the norms defined by a self-scaled barrier.

LEMMA 3.2. a) Let $x \in \text{int } K$ and $s \in \text{int } K^*$. Then for any values of τ and ξ we have

$$(3.3) \quad \|\tau x + \xi F'_*(s)\|_x = \|\tau s + \xi F'(x)\|_s,$$

$$(3.4) \quad \|\tau x + \xi F'_*(s)\|_s = \|\tau s + \xi F'(x)\|_x.$$

b) For any $x, v \in \text{int } K$

$$\|v\|_x = \|F'(x)\|_v.$$

Proof. For part a), let us choose $w \in K$ such that $s = F''(w)x$ (see T_* 3.2). Then

$$F'(x) = F''(w)F'_*(s), \quad F''(x) = F''(w)F''_*(s)F''(w).$$

Also, let $v \in E$ and $t := F''(w)v \in E^*$. Then we obtain

$$\|v\|_x^2 = \langle F''(x)v, v \rangle = \langle F''(w)F''_*(s)F''(w)v, v \rangle = \langle t, F''_*(s)t \rangle = \|t\|_s^2.$$

In particular, we can take $v = \tau x + \xi F'_*(s)$ and then $t = \tau s + \xi F'(x)$. This proves the first equation; the second is proved similarly. Part b) follows from the first equation by taking $s = -F'(v)$, $\tau = 0$, and $\xi = 1$. $\square$

For semidefinite programming, (3.3) states that the 2-norm of the vector of eigenvalues of $\tau i - \xi x^{-1/2} s^{-1} x^{-1/2} = x^{-1/2}(\tau x - \xi s^{-1})x^{-1/2}$ is the same as for $\tau i - \xi s^{-1/2} x^{-1} s^{-1/2} = s^{-1/2}(\tau s - \xi x^{-1})s^{-1/2}$; (3.7) says the same for the infinity-norm (here i denotes the identity). This is clear, since the two matrices are similar and hence have the same eigenvalues. To see the steps of the proof in this setting, recall that, in this case, the point w is $s^{-1/2}(s^{1/2}x s^{1/2})^{1/2}s^{-1/2} = x^{1/2}(x^{1/2} s x^{1/2})^{-1/2}x^{1/2}$.

For $\tau = 1$ and $\xi = 0$, part a) of the lemma yields (4.1)*. Note that, for w as above, we also have $\|\tau s + \xi F'(x)\|_w = \|\tau x + \xi F'_*(s)\|_w$.

LEMMA 3.3. *Let $x \in \text{int } K$ and $s \in \text{int } K^*$. Then for any values of τ and ξ we have*

$$(3.5) \quad \sigma_x(\tau x + \xi F'_*(s)) = \sigma_s(\tau s + \xi F'(x)),$$

$$(3.6) \quad \sigma_s(\tau x + \xi F'_*(s)) = \sigma_x(\tau s + \xi F'(x)),$$

$$(3.7) \quad |\tau x + \xi F'_*(s)|_x = |\tau s + \xi F'(x)|_s,$$

$$(3.8) \quad |\tau x + \xi F'_*(s)|_s = |\tau s + \xi F'(x)|_x.$$

Proof. Again choose $w \in K$ such that $s = F''(w)x$ (see T_* 3.2). Then $F'(x) = F''(w)F'_*(s)$. Then in view of T_* 3.1 (iii), the inclusion

$$x - \alpha(\tau x + \xi F'_*(s)) \in K$$

is true for some $\alpha \geq 0$ if and only if

$$s - \alpha(\tau s + \xi F'(x)) \in K^*.$$

This proves (3.5); the proof of the (3.6) is similar. Then (3.7) and (3.8) follow from these. $\square$

Again, for w as above, we have also $|\tau s + \xi F'(x)|_w = |\tau x + \xi F'_*(s)|_w$.

Let us now prove some relations among the σ -measures for x , s , and the scaling point w . (The result below is referred to in the penultimate paragraph of section 8 of [NT97].)

LEMMA 3.4. *Let $x \in \text{int } K$ and $s \in \text{int } K^*$, and define w as the unique point in $\text{int } K$ such that $F''(w)x = s$. Then,*

$$\sigma_s(-F'(x)) = \sigma_x(-F'_*(s)) = \sigma_x(w)^2 = \sigma_w(-F'(x))^2 = \sigma_s(-F'(w))^2 = \sigma_w(-F'_*(s))^2,$$

$$\sigma_s(x) = \sigma_x(s) = \sigma_w(x)^2 = \sigma_x(-F'(w))^2 = \sigma_w(s)^2 = \sigma_s(w)^2.$$

Proof. Let us prove the first sequence of identities. The equality of $\sigma_s(-F'(x))$ and $\sigma_x(-F'_*(s))$ follows from Lemma 3.3, as does that of $\sigma_x(w)$ and $\sigma_w(-F'(x))$ and that of $\sigma_s(-F'(w))$ and $\sigma_w(-F'_*(s))$. Moreover, the proof of $\sigma_x(w) = \sigma_s(-F'(w))$ is exactly analogous. It remains to show that $\sigma_x(-F'_*(s)) = \sigma_x(w)^2$. Let us write

$z := -F'_*(s)$ and abbreviate $\sigma_x(w)$ to σ . Since x and w lie in $\text{int } K$, σ is positive and finite, and

$$\sigma x - w \in \partial K.$$

We want to show that

$$\sigma^2 x - z = \sigma(\sigma x - w) + (\sigma w - z) \in \partial K$$

also.

We saw above that $\sigma = \sigma_w(z)$, so $\sigma w - z \in \partial K$ and thus $\sigma^2 x - z \in K$. We need to show that it lies in the boundary of K . Because $\sigma x - w \in \partial K$, there is some $u \in \partial K^*$ with $\langle u, \sigma x - w \rangle = 0$. Let $v := [F''(w)]^{-1}u \in \partial K$, so that

$$\langle F''(w)v, \sigma x - w \rangle = 0,$$

i.e., v and $\sigma x - w$ are orthogonal with respect to w (see section 5 of [NT97]). We aim to show that v is also orthogonal to $\sigma w - z$ with respect to w , so that it is orthogonal to $\sigma^2 x - z$ and thus the latter lies in ∂K .

By L_* 5.1, v and $\sigma x - w$ are also orthogonal with respect to $w(\alpha) := w + \alpha(\sigma x - w)$ for any $\alpha \geq 0$. Hence, we find $\langle F''(w(\alpha))(\sigma x - w), v \rangle = 0$ for all $\alpha \geq 0$, which implies

$$\langle F'(\sigma x) - F'(w), v \rangle = \left\langle \int_0^1 F''(w(\alpha))(\sigma x - w) d\alpha, v \right\rangle = 0.$$

This gives

$$\begin{aligned} \langle F''(w)v, \sigma w - z \rangle &= \langle F''(w)v, \sigma w + F'_*(s) \rangle \\ &= \langle F''(w)v, [F''(w)]^{-1}(F'(x) - \sigma F'(w)) \rangle \\ &= \sigma \langle F'(\sigma x) - F'(w), v \rangle = 0, \end{aligned}$$

where the last equation used (2.7). Hence, $\sigma^2 x - z = \sigma(\sigma x - w) + (\sigma w - z)$ is orthogonal to v with respect to w , and thus lies in ∂K as desired.

The proof of the second line of identities is similar. $\square$

In the semidefinite programming case, the lemma says that the maximum eigenvalue of $s^{-1/2}x^{-1}s^{-1/2}$ is equal to that of $x^{-1/2}s^{-1}x^{-1/2}$, and that this is the square of the maximum eigenvalue of $x^{-1/2}wx^{-1/2} = (x^{1/2}sx^{1/2})^{-1/2}$.

3.2. Inequalities for norms. We have the following relationship between the two norms defined at a point.

PROPOSITION 3.5. *For $a \in E$ or E^* and for $b \in \text{int } K$ or $\text{int } K^*$ we have*

$$\|a\|_b \leq \|a\|_b \nu^{1/2} \|a\|_b.$$

Proof. Since all the cases are similar, we assume $a \in E$ and $b \in \text{int } K$. The first inequality follows from (4.5)*. For the second, suppose that $b + \alpha a$ and $b - \alpha a \in K$ for some $\alpha > 0$. Then,

$$\nu - \alpha^2 \|a\|_b^2 = \langle F''(b)b, b \rangle - \alpha^2 \langle F''(b)a, a \rangle = \langle F''(b)(b - \alpha a), b + \alpha a \rangle \geq 0,$$

so that $\|a\|_b \leq \nu^{1/2}/\alpha$. Hence, this holds for the supremum of all such α 's, which gives the right-hand inequality. $\square$

We now give a result that relates the norms at two neighboring points. (Compare with Theorem 3.8 below.) Note that the outer bounds in (3.9) and (3.10) are valid for general self-concordant functions under the much stronger assumption that $\|\tilde{x} - x\|_x \leq \delta < 1$ (see T^* 2.1.1).

THEOREM 3.6. *Suppose x and $\tilde{x}$ lie in $\text{int } K$ with $\|\tilde{x} - x\|_x \leq \delta < 1$. Also, let $p := x - \tilde{x}$, so that $p_+ := \sigma_x(p) \leq \delta$ and $p_- := \sigma_x(-p) \leq \delta$. Then, for any $v \in E$ and $u \in E^*$, we have*

$$(3.9) \quad \begin{aligned} (1 + \delta)^{-1} \|v\|_x &\leq (1 + p_-)^{-1} \|v\|_x \leq \|v\|_{\tilde{x}} \\ &\leq (1 - p_+)^{-1} \|v\|_x \leq (1 - \delta)^{-1} \|v\|_x, \end{aligned}$$

$$(3.10) \quad \begin{aligned} (1 - \delta) \|u\|_x &\leq (1 - p_+) \|u\|_x \leq \|u\|_{\tilde{x}} \\ &\leq (1 + p_-) \|u\|_x \leq (1 + \delta) \|u\|_x, \end{aligned}$$

$$(3.11) \quad \begin{aligned} (1 + \delta)^{-1} |v|_x &\leq (1 + p_-)^{-1} |v|_x \leq |v|_{\tilde{x}} \\ &\leq (1 - p_+)^{-1} |v|_x \leq (1 - \delta)^{-1} |v|_x, \end{aligned}$$

$$(3.12) \quad \begin{aligned} (1 - \delta) |u|_x &\leq (1 - p_-) |u|_x \leq |u|_{\tilde{x}} \\ &\leq (1 + p_+) |u|_x \leq (1 + \delta) |u|_x. \end{aligned}$$

Proof. The inequalities in (3.9) and (3.10) follow directly from T_* 4.1. For (3.11), suppose first that $0 < \alpha < 1/\sigma_x(v)$. Then $x - \alpha v$ lies in K , and hence,

$$(1 - p_+)x - \alpha(1 - p_+)v \in K.$$

Also,

$$-(1 - p_+)x + \tilde{x} = p_+x - (x - \tilde{x}) \in K,$$

so that $\tilde{x} - \alpha(1 - p_+)v \in K$ and $1/\sigma_{\tilde{x}}(v) \geq \alpha(1 - p_+)$. Since $0 < \alpha < 1/\sigma_x(v)$ was arbitrary, we have $\sigma_{\tilde{x}}(v) \leq (1 - p_+)^{-1}\sigma_x(v)$, and because this holds also for $-v$, we conclude that

$$|v|_{\tilde{x}} \leq (1 - p_+)^{-1} |v|_x.$$

Similarly, if $0 < \alpha < 1/\sigma_{\tilde{x}}(v)$, then $\tilde{x} - \alpha v$, and hence,

$$(1 + p_-)^{-1}\tilde{x} - \alpha(1 + p_-)^{-1}v \in K.$$

Also,

$$(1 + p_-)x - \tilde{x} = p_-x + (x - \tilde{x}) \in K,$$

so $x - (1 + p_-)^{-1}\tilde{x} \in K$ and thus $x - \alpha(1 + p_-)^{-1}v \in K$. We conclude that $1/\sigma_x(v) \geq \alpha(1 + p_-)^{-1}$ and hence, $\sigma_{\tilde{x}}(v) \geq (1 + p_-)^{-1}\sigma_x(v)$. Since this holds also for $-v$, we obtain

$$|v|_{\tilde{x}} \geq (1 + p_-)^{-1} |v|_x,$$

finishing the proof of (3.11).

Now (3.5) in Lemma 3.3 with $s := -F'(\tilde{x})$ shows that

$$\sigma_{\tilde{x}}(F'(x) - F'(\tilde{x})) = p_+, \quad \sigma_{\tilde{x}}(-F'(x) + F'(\tilde{x})) = p_-,$$

so (3.12) follows from (3.11) by replacing v , x , and $\tilde{x}$ with u , $-F'(\tilde{x})$, and $-F'(x)$ respectively. $\square$

3.3. Properties of the third derivative. Let us now establish some properties of the third derivative of the self-scaled barrier. Let $x \in \text{int } K$ and $p_1, p_2, p_3 \in E$. Then,

$$F'''(x)[p_1, p_2, p_3] = (\langle F''(x + \alpha p_3)p_1, p_2 \rangle)'_{\alpha=0}.$$

Note that the third derivative is a symmetric trilinear form. We denote by $F'''(x)[p_1, p_2]$ a vector from E^* such that

$$F'''(x)[p_1, p_2, p_3] = \langle F'''(x)[p_1, p_2], p_3 \rangle$$

for any $p_3 \in E$. Similarly, $F'''(x)[p_1]$ is a linear operator from E to E^* such that

$$F'''(x)[p_1, p_2, p_3] = \langle F'''(x)[p_1]p_2, p_3 \rangle$$

for any $p_2, p_3 \in E$.

In the semidefinite programming case, $F'''(x)[p_1, p_2]$ is the matrix $-x^{-1}p_1x^{-1}p_2x^{-1} - x^{-1}p_2x^{-1}p_1x^{-1}$.

The results below are based on the following important property.

LEMMA 3.7. *For any $x \in \text{int } K$ and $p \in E$*

$$(3.13) \quad \| F'''(x)[p] \|_x \leq 2 \| p \|_x.$$

Proof. Without loss of generality, we can assume that $\| p \|_x = 1$. Then $x + p \in K$ and $x - p \in K$.

Note that in view of C_* 3.2 (i), the operator $F'''(x)[v]$ is negative semidefinite for any $v \in K$. Therefore,

$$F'''(x)[p] = F'''(x)[(x + p) - x] = 2F''(x) + F'''(x)[x + p] \leq 2F''(x).$$

Similarly,

$$F'''(x)[p] = F'''(x)[x - (x - p)] = -2F''(x) - F'''(x)[x - p] \geq -2F''(x)$$

and (3.13) follows. $\square$

Using the lemma above we can obtain some useful inequalities.

THEOREM 3.8. *For any $x \in \text{int } K$ and $p_1, p_2 \in E$ the following inequalities hold:*

$$(3.14) \quad \| F'''(x)[p_1, p_2] \|_x \leq 2 \| p_1 \|_x \cdot \| p_2 \|_x.$$

Moreover, if $v = x - p_1 \in \text{int } K$ then,

$$(3.15) \quad \| (F''(v) - F''(x))p_2 \|_x \leq \frac{2 - \sigma_x(p_1)}{(1 - \sigma_x(p_1))^2} \| p_1 \|_x \cdot \| p_2 \|_x.$$

Proof. The proof of (3.14) immediately follows from (3.13) and (3.1). Let us prove (3.15).

In view of (3.14), Theorem 3.6, and T_* 4.1 we have

$$\begin{aligned}
\| (F''(v) - F''(x)) p_2 \|_x &= \left\| \int_0^1 F'''(x - \tau p_1) [p_1, p_2] d\tau \right\|_x \leq \int_0^1 \| F'''(x - \tau p_1) [p_1, p_2] \|_x d\tau \\
&\leq \int_0^1 \frac{\| F'''(x - \tau p_1) [p_1, p_2] \|_{x - \tau p_1}}{1 - \tau \sigma_x(p_1)} d\tau \leq \int_0^1 \frac{2}{1 - \tau \sigma_x(p_1)} \| p_1 \|_{x - \tau p_1} \cdot \| p_2 \|_{x - \tau p_1} d\tau \\
&\leq \| p_1 \|_x \cdot \| p_2 \|_x \int_0^1 \frac{2 d\tau}{(1 - \tau \sigma_x(p_1))^3} = \frac{2 - \sigma_x(p_1)}{(1 - \sigma_x(p_1))^2} \| p_1 \|_x \cdot \| p_2 \|_x. \quad \square
\end{aligned}$$

Note that from (3.14) for any $p_3 \in E$ we have

$$(3.16) \quad \| F'''(x) [p_1, p_2, p_3] \| \leq 2 \| p_1 \|_x \cdot \| p_2 \|_x \cdot \| p_3 \|_x.$$

This inequality highlights the difference between self-scaled barriers and arbitrary self-concordant functions; see D^* 2.1.1 and its seemingly more general consequence in T^* 2.1.1, which is (3.16) but with $\| p_1 \|_x$ replaced by $\| p_1 \|_x$.

We will also need a kind of “parallelogram rule” for the self-scaled barrier. Let us first prove two simple lemmas.

LEMMA 3.9. *Let $x, v \in \text{int } K$. Then,*

$$(3.17) \quad [F''(v)]^{-1} F'(x) = \frac{1}{2} [F''(x)]^{-1} F'''(x) [v, v].$$

Proof. Let $s = -F'(v)$. Then, in view of Lemma 3.2 a), we have

$$\langle F''_*(s) F'(x), F'(x) \rangle \equiv \langle F''(x) F'_*(s), F'_*(s) \rangle.$$

By differentiating this identity with respect to x we get

$$2F''(x) F''_*(s) F'(x) \equiv F'''(x) [F'_*(s), F'_*(s)],$$

and that is exactly (3.17) since

$$F'_*(s) = -v, \quad F''_*(s) = [F''(v)]^{-1}$$

using (2.15) and (2.16). $\square$

LEMMA 3.10. *Let $x \in \text{int } K$ and $s \in \text{int } K^*$. Then,*

$$(3.18) \quad F_*(s + \tau F'(x)) + F(x) \equiv F_*(s) + F(x + \tau F'_*(s))$$

for any τ such that $s + \tau F'(x) \in \text{int } K^*$.

Proof. We simply note that

$$s + \tau F'(x) = F''(w)(x + \tau F'_*(s)),$$

where $w \in \text{int } K$ is such that $s = F''(w)x$. Therefore, (3.18) immediately follows from (2.19). $\square$

(We note that this lemma was proved for the universal barrier of K by Rothaus in Theorem 3.8 of [Ro60].)

Now we can prove the following important theorem.

THEOREM 3.11. *Let $x \in \text{int } K$, $p \in E$, and the positive constants α and β be such that $x - \alpha p \in \text{int } K$ and $x + \beta p \in \text{int } K$. Then,*

$$F(x - \alpha p) + F(x + \beta p) = F(x) + F\left(x + (\beta - \alpha)p + \frac{1}{2}\alpha\beta[F''(x)]^{-1}F'''(x)[p, p]\right).$$

Proof. Denote

$$v = x + (\beta - \alpha)p + \frac{1}{2}\alpha\beta[F''(x)]^{-1}F'''(x)[p, p], \quad x_1 = x - \alpha p, \quad x_2 = x + \beta p.$$

Then in view of (2.8) and Lemma 3.9,

$$\begin{aligned} v &= x + (\beta - \alpha)p + \frac{\alpha}{2}[F''(x)]^{-1}F'''(x)[p, x_2 - x] \\ &= x_2 + \frac{\alpha}{2}[F''(x)]^{-1}F'''(x)[p, x_2] \\ &= x_2 + \frac{\alpha}{2\beta}[F''(x)]^{-1}F'''(x)[(x_2 - x, x_2)] \\ &= \left(1 + \frac{\alpha}{\beta}\right)x_2 + \frac{\alpha}{2\beta}[F''(x)]^{-1}F'''(x)[x_2, x_2] \\ &= \left(1 + \frac{\alpha}{\beta}\right)x_2 + \frac{\alpha}{\beta}[F''(x_2)]^{-1}F'(x) \\ &= [F''(x_2)]^{-1}\left[-\left(1 + \frac{\alpha}{\beta}\right)F'(x_2) + \frac{\alpha}{\beta}F'(x)\right]. \end{aligned}$$

Now, let $s = -F'(x_2)$. Then, in view of (2.20), (2.6), and Lemma 3.10 we have

$$\begin{aligned} F(v) &= F\left([F''(x_2)]^{-1}\left[\left(1 + \frac{\alpha}{\beta}\right)s + \frac{\alpha}{\beta}F'(x)\right]\right) \\ &= F_*\left(\left(1 + \frac{\alpha}{\beta}\right)s + \frac{\alpha}{\beta}F'(x)\right) + 2F(x_2) + \nu \\ &= F_*\left(s + \frac{\alpha}{\alpha + \beta}F'(x)\right) + 2F(x_2) + \nu + \nu \ln \frac{\beta}{\alpha + \beta} \\ &= -F(x) + F_*(s) + F\left(x - \frac{\alpha}{\alpha + \beta}x_2\right) + 2F(x_2) + \nu + \nu \ln \frac{\beta}{\alpha + \beta} \\ &= -F(x) + F\left(\frac{\beta x - \alpha \beta p}{\alpha + \beta}\right) + F(x_2) + \nu \ln \frac{\beta}{\alpha + \beta} \\ &= -F(x) + F(x_1) + F(x_2). \end{aligned}$$

Of course, here we have assumed that the appropriate points lie in the domains of F and F_* . But $x_1 = x - \alpha p \in \text{int } K$ implies

$$x - \frac{\alpha}{\alpha + \beta}x_2 = \frac{\beta x - \alpha \beta p}{\alpha + \beta} \in \text{int } K.$$

Hence, using w with $F''(w)x = s$, we find

$$s + \frac{\alpha}{\alpha + \beta}F'(x) \text{ and hence } \left(1 + \frac{\alpha}{\beta}\right)s + \frac{\alpha}{\beta}F'(x) \in \text{int } K^*.$$

This shows $v \in \text{int } K$, and hence, all the terms above are well defined. $\square$

3.4. Some properties of the scaling point. In this section we prove some relations for $x \in \text{int } K$, $s \in \text{int } K^*$, and the scaling point $w \in \text{int } K$ such that $F''(w)x = s$.

Denote by $\mu(x, s)$ the *normalized duality gap*:

$$\mu(x, s) := \frac{1}{\nu} \langle s, x \rangle.$$

LEMMA 3.12. *Let $\bar{w} := \sqrt{\mu(x, s)}w$, $\bar{x} := x/\sqrt{\mu(x, s)}$. Then,*

$$(3.19) \quad 2(F(x) - F(\bar{w})) \equiv F(x) + F_*(s) + \nu \ln \mu(x, s) + \nu \geq 0,$$

$$(3.20) \quad 2[F(x) - F(\bar{w})] \geq \|\bar{x} - w\|_w^2 = \|\bar{w} - x\|_{\bar{w}}^2,$$

$$(3.21) \quad 2[F(x) - F(\bar{w})] \leq [\mu(x, s) \langle F'(x), F'_*(s) \rangle - \nu] - \|\bar{w} - x\|_x^2.$$

Proof. Denote $\mu = \mu(x, s)$. Then,

$$F(\bar{w}) = F(w) - \frac{1}{2} \nu \ln \mu = \frac{1}{2} [F(x) - F_*(s) - \nu - \nu \ln \mu].$$

(We have used (2.6) and (2.19).) Therefore, in view of (2.17), we obtain

$$2(F(x) - F(\bar{w})) = F(x) + F_*(s) + \nu \ln \mu + \nu \geq 0.$$

Further, from (2.9) and the convexity of F , we have

$$\begin{aligned} \|\bar{x} - w\|_w^2 &= \frac{1}{\mu} \langle s, x \rangle + \frac{2}{\sqrt{\mu}} \langle F'(w), x \rangle + \nu \\ &= 2[\nu + \langle F'(\bar{w}), x \rangle] = 2\langle F'(\bar{w}), x - \bar{w} \rangle \leq 2(F(x) - F(\bar{w})), \end{aligned}$$

which gives (3.20); the equation is trivial by (2.7).

Finally, note that in view of Lemma 3.2(b), we have

$$\begin{aligned} \|\bar{w} - x\|_x^2 &= \mu \langle F''(x)w, w \rangle + 2\langle F'(x), \bar{w} \rangle + \nu \\ &= \mu \langle F'(x), [F''(w)]^{-1} F'(x) \rangle + 2\langle F'(x), \bar{w} \rangle + \nu \\ &= [\mu \langle F'(x), F'_*(s) \rangle - \nu] + 2(\langle F'(x), \bar{w} \rangle + \nu). \end{aligned}$$

Therefore, using the convexity of $F(x)$, we get

$$2(F(x) - F(\bar{w})) \leq 2\langle F'(x), x - \bar{w} \rangle = -2(\langle F'(x), \bar{w} \rangle + \nu),$$

and (3.21) follows. $\square$

Let $t := -F'(w)$, so that $F''(t) = [F''(w)]^{-1}$ takes s into x . We next relate $F''(w)$ to $F''(x)$ and $F''_*(t)$ to $F''_*(s)$.

LEMMA 3.13. *Suppose we have $\delta := |s/\mu + F'(x)|_x < 1$ for some $\mu > 0$. Then,*

$$(3.22) \quad \begin{aligned} (1 - \delta)F''(x) &\leq \frac{F''(w)}{\mu} \leq (1 + \delta)F''(x), \\ (1 - \delta)F''_*(s) &\leq \frac{F''_*(t)}{\mu} \leq (1 + \delta)F''_*(s). \end{aligned}$$

Proof. Let $v := -F'_*(s)$ and $u := -F'(x)$. We will use C_* 4.1 to prove our result, so we need bounds on $\sigma_v(x)$, $\sigma_x(v)$, $\sigma_u(s)$, and $\sigma_s(u)$. Since $x, v \in \text{int } K$ and $s, u \in \text{int } K^*$, these are all positive, and we have using Lemma 3.3

$$\begin{aligned} [\sigma_v(x)]^{-1} &= [\sigma_u(s)]^{-1} \\ &= \sup\{\alpha : -F'(x) - \alpha s \in K^*\} \\ &= \sup\left\{\alpha : (1 - \alpha\mu)(-F'(x)) - \alpha\mu\left(\frac{s}{\mu} + F'(x)\right) \in K^*\right\}. \end{aligned}$$

Now for $\alpha < [\mu(1 + \delta)]^{-1}$, $\alpha\mu < 1$ and $[\alpha\mu/(1 - \alpha\mu)]\delta < 1$, so using $\delta := \|s/\mu + F'(x)\|_x$ we have $-F'(x) - [\alpha\mu/(1 - \alpha\mu)](s/\mu + F'(x)) \in K^*$. Thus $[\sigma_v(x)]^{-1} \geq [\mu(1 + \delta)]^{-1}$, or

$$(3.23) \quad \sigma_v(x) = \sigma_u(s) \leq \mu(1 + \delta).$$

Similarly,

$$\begin{aligned} [\sigma_x(v)]^{-1} &= [\sigma_s(u)]^{-1} \\ &= \sup\{\alpha : s + \alpha F'(x) \in K\} \\ &= \sup\{\alpha : (\mu - \alpha)(-F'(x)) + \mu\left(\frac{s}{\mu} + F'(x)\right) \in K^*\}. \end{aligned}$$

Now for $\alpha < \mu(1 - \delta)$, $\alpha < \mu$ and $[\mu/(\mu - \alpha)]\delta < 1$, so using $\delta := \|s/\mu + F'(x)\|_x$ again we have $-F'(x) + [\mu/(\mu - \alpha)](s/\mu + F'(x)) \in K^*$. Thus, $[\sigma_x(v)]^{-1} \geq \mu(1 - \delta)$ or

$$(3.24) \quad \sigma_x(v) = \sigma_s(u) \leq [\mu(1 - \delta)]^{-1}.$$

Then (3.23) and (3.24) imply (3.22) by C_* 4.1(iii). $\square$

COROLLARY 3.14. *Under the conditions of the lemma, for all $v \in E$, $u \in E^*$, we have*

$$(3.25) \quad \left\| \left[\frac{F''(w)}{\mu} - F''(x) \right] v \right\|_x \leq \delta \|v\|_x,$$

$$(3.26) \quad \left\| \left[\frac{F''_*(t)}{\mu} - F''_*(s) \right] u \right\|_s \leq \delta \|u\|_s.$$

Proof. This follows from the lemma and (3.2). $\square$

4. Primal-dual central path and proximity measures. In this section we consider *general* primal-dual proximity measures. These measures estimate the distance from a primal-dual pair $(x, s) \in \text{int } K \times \text{int } K^*$ to the nonlinear surface of “analytic centers”

$$\mathcal{A} := \{(x, s) \in \text{int } K \times \text{int } K^* : s = -\mu F'(x), \mu > 0\}.$$

(The intersection of the Cartesian product of this surface with Y and the strictly feasible set of the primal-dual problem (2.1)–(2.3), $S^0(P) \times S^0(D)$, is called the *primal-dual central path*.) Note that these general proximity measures do not at all involve the specific information about our problem (2.1), namely, the objective vector c , the operator A , and the right-hand side b .

Let us first introduce the notion of the central path and give the main properties of this trajectory. In what follows, we assume that Assumptions (2.4) and (2.5) are satisfied.

Define the *primal central path* $\{x(\mu) : \mu > 0\}$ for the problem (2.1) as follows:

$$(4.1) \quad x(\mu) = \arg \min_x \left\{ \frac{1}{\mu} \langle c, x \rangle + F(x) : Ax = b, x \in K \right\}.$$

The *dual central path* $\{(y(\mu), s(\mu)) : \mu > 0\}$ for the dual problem (2.3) is defined in a symmetric way:

$$(4.2) \quad (y(\mu), s(\mu)) = \arg \min_{y,s} \left\{ -\frac{1}{\mu} \langle b, y \rangle + F_*(s) : A^*y + s = c, y \in Y, s \in K^* \right\}.$$

The *primal-dual central path* for the primal-dual problem

$$(4.3) \quad \begin{aligned} \min \quad & \langle c, x \rangle - \langle b, y \rangle, \\ \text{s.t.} \quad & Ax = b, \\ & A^*y + s = c, \\ & x \in K, y \in Y, s \in K^*, \end{aligned}$$

is simply the combination of these paths:

$$\{v(\mu) = (x(\mu), y(\mu), s(\mu)) : \mu > 0\}.$$

Let us present now the duality theorem (see [Ne96]).

THEOREM 4.1.

a) For any $\xi > 0$ the set

$$Q(\xi) = \{v = (x, y, s) : Ax = b, A^*y + s = c, \langle c, x \rangle - \langle b, y \rangle = \xi, x \in K, y \in Y, s \in K^*\}$$

is nonempty and bounded. The optimal set of the problem (4.3) is also nonempty and bounded.

b) The optimal value of the problem (4.3) is zero.

c) The points of the primal-dual central path are well defined and for any $\mu > 0$ the following relations hold:

$$(4.4) \quad \langle c, x(\mu) \rangle - \langle b, y(\mu) \rangle = \langle s(\mu), x(\mu) \rangle = \nu\mu,$$

$$(4.5) \quad F(x(\mu)) + F_*(s(\mu)) = -\nu - \nu \ln \mu,$$

$$(4.6) \quad s(\mu) = -\mu F'(x(\mu)),$$

$$(4.7) \quad x(\mu) = -\mu F'_*(s(\mu)). \quad \square$$

For linear programming, where K is the nonnegative orthant in $E := R^n$, (4.6) and (4.7) state that the componentwise product of x and s is μ times the vector of ones in R^n . For semidefinite programming, where K is the cone of positive semidefinite matrices in the space E of symmetric matrices of order n , they state that the matrix product of x and s is μ times the identity matrix of order n .

The relations of c) of the theorem explain the construction of all general primal-dual proximity measures: most of them measure in a specific way the residual

$$\frac{1}{\mu} s + F'(x)$$

with $\mu = \mu(x, s)$, where

$$\mu(x, s) := \frac{1}{\nu} \langle s, x \rangle.$$

In our paper we will consider two groups of primal-dual proximity measures. The first group consists of the following *global* proximity measures:

- the *functional* proximity measure

$$(4.8) \quad \gamma_F(x, s) := F(x) + F_*(s) + \nu \ln \mu(x, s) + \nu,$$

- the *gradient* proximity measure

$$(4.9) \quad \gamma_G(x, s) := \mu(x, s) \langle F'(x), F'_*(s) \rangle - \nu,$$

- the *uniform* proximity measure

$$(4.10) \quad \gamma_\infty(x, s) := \mu(x, s) \sigma_s(-F'(x)) - 1 = \mu(x, s) \sigma_x(-F'_*(s)) - 1$$

(see Lemma 3.3).

The second group is comprised of the *local* proximity measures:

$$(4.11) \quad \lambda_\infty(x, s) := \left| \frac{1}{\mu(x, s)} s + F'(x) \right|_x = \left| \frac{1}{\mu(x, s)} x + F'_*(s) \right|_s$$

(see Lemma 3.3),

$$(4.12) \quad \lambda_\infty^+(x, s) := \frac{\sigma_s(x)}{\mu(x, s)} - 1 = \frac{\sigma_x(s)}{\mu(x, s)} - 1$$

(see Lemma 3.3 and compare with (4.10)),

$$(4.13) \quad \lambda_2(x, s) := \left\| \frac{1}{\mu(x, s)} s + F'(x) \right\|_x = \left\| \frac{1}{\mu(x, s)} x + F'_*(s) \right\|_s$$

(see Lemma 3.2(a)), and

$$(4.14) \quad \lambda(x, s) := \left[\nu - \frac{\nu^2 \mu^2(x, s)}{\|s\|_x^2} \right]^{1/2} = \left[\nu - \frac{\nu^2 \mu^2(x, s)}{\|x\|_s^2} \right]^{1/2}$$

(see Lemma 3.2 a)). The proof of Theorem 4.2 shows that $\lambda(x, s)$ is well defined. We will later see that the value of any global proximity measure can be used as an estimate for the distance to the primal-dual central path at any strictly feasible primal-dual point. The value of a local proximity measure has a sense only in a small neighborhood of the central path.

For comparison with the standard proximity measures developed in linear and semidefinite programming, let us present the specific form of the above proximity

measures for the cases $K = R_+^n$ and $K = S_+^n$, where i denotes the identity matrix:

$$\begin{aligned}
\gamma_F(x, s) &= -\sum_{j=1}^n \ln(x^{(j)} s^{(j)}) + n \ln\left(\frac{x^T s}{n}\right), \\
&\quad -\ln \det x - \ln \det s + n \ln \frac{\text{Trace}(x^{1/2} s x^{1/2})}{n}, \\
\gamma_G(x, s) &= \frac{x^T s}{n} \sum_{j=1}^n \frac{1}{x^{(j)} s^{(j)}} - n, \\
&\quad \frac{\text{Trace}(x^{1/2} s x^{1/2}) \text{Trace}(x^{-1/2} s^{-1} x^{-1/2})}{n} - n, \\
\gamma_\infty(x, s) &= \frac{x^T s}{n \min_{1 \leq j \leq n} x^{(j)} s^{(j)}} - 1, \\
&\quad \frac{\text{Trace}(x^{1/2} s x^{1/2})}{n \chi_{\min}(x^{1/2} s x^{1/2})} - 1, \\
\lambda_\infty(x, s) &= \left\| \frac{n}{x^T s} X s - e \right\|_\infty, \\
&\quad \left\| \frac{x^{1/2} s x^{1/2}}{\text{Trace}(x^{1/2} s x^{1/2})/n} - i \right\|_2, \\
\lambda_\infty^+(x, s) &= \frac{n}{x^T s} \max_{1 \leq j \leq n} x^{(j)} s^{(j)} - 1, \\
&\quad \chi_{\max} \left(\frac{x^{1/2} s x^{1/2}}{\text{Trace}(x^{1/2} s x^{1/2})/n} - i \right), \\
\lambda_2(x, s) &= \left\| \frac{n}{x^T s} X s - e \right\|_2, \\
&\quad \left\| \frac{x^{1/2} s x^{1/2}}{\text{Trace}(x^{1/2} s x^{1/2})/n} - i \right\|_F, \\
\lambda(x, s) &= \left[n - \frac{\left(\sum_{j=1}^n x^{(j)} s^{(j)} \right)^2}{\sum_{j=1}^n (x^{(j)} s^{(j)})^2} \right]^{1/2}, \\
&\quad \left[n - \frac{[\text{Trace}(x^{1/2} s x^{1/2})]^2}{\|x^{1/2} s x^{1/2}\|_F^2} \right]^{1/2}.
\end{aligned}$$

Thus level sets of λ_2 , λ_∞ , and γ_∞ correspond respectively to the 2-norm, infinity-norm, and “minus-infinity-norm” neighborhoods of the central path used in interior-point methods for linear programming. Also, γ_F is the Tanabe–Todd–Ye primal-dual potential function with parameter n . (We have written these measures for semidefinite programming in a symmetric way that stresses the similarity to linear programming. However, simpler formulae should be used for computation; for example, $\text{Trace}(x^{1/2} s x^{1/2}) = \text{Trace}(xs)$, and the eigenvalues of $x^{1/2} s x^{1/2}$ are the same as those of $l^T s l$, where $l l^T$ is the Cholesky factorization of x .)

Now we return to the general case.

Remark 1. Note that the gradient proximity measure can be written as a squared norm of the vector $s/\mu(x, s) + F'(x)$. Indeed, let us choose $w \in \text{int } K$ such that $s = F''(w)x$ (this is possible in view of T_* 3.2). Then $F'(x) = F''(w)F'_*(s)$. Therefore (writing μ for $\mu(x, s)$ for simplicity),

$$\begin{aligned} & \left\langle \frac{1}{\mu}s + F'(x), [F''(w)]^{-1} \left(\frac{1}{\mu}s + F'(x) \right) \right\rangle \\ &= \frac{1}{\mu^2} \langle s, [F''(w)]^{-1}s \rangle + \frac{2}{\mu} \langle F'(x), [F''(w)]^{-1}s \rangle + \langle F'(x), [F''(w)]^{-1}F'(x) \rangle \\ &= \frac{\nu}{\mu} - \frac{2\nu}{\mu} + \langle F'(x), F'_*(s) \rangle = \frac{\gamma_G(x, s)}{\mu}. \end{aligned}$$

Thus,

$$(4.15) \quad \gamma_G(x, s) = \mu(x, s) \left\| \frac{1}{\mu(x, s)}s + F'(x) \right\|_w^2 = \mu(x, s) \left\| \frac{1}{\mu(x, s)}x + F'_*(s) \right\|_w^2.$$

Remark 2. There is an interesting relation between λ_∞ , λ_∞^+ , and γ_∞ :

$$\lambda_\infty(x, s) = \max \left\{ \lambda_\infty^+(x, s), \frac{\gamma_\infty(x, s)}{1 + \gamma_\infty(x, s)} \right\}.$$

Note that all the proximity measures we have introduced are homogeneous of degree zero (indeed, separately in x and s). Let us prove some relations between these measures.

THEOREM 4.2. *Let $x \in \text{int } K$ and $s \in \text{int } K^*$. Then,*

$$(4.16) \quad \lambda_\infty^+(x, s) \leq \lambda_\infty(x, s) \leq \lambda_2(x, s) \leq \nu^{1/2} \lambda_\infty(x, s),$$

$$(4.17) \quad \gamma_F(x, s) \leq \nu \ln \left(1 + \frac{1}{\nu} \gamma_G(x, s) \right) \leq \gamma_G(x, s),$$

$$(4.18) \quad \frac{\gamma_\infty^2(x, s)}{1 + \gamma_\infty(x, s)} \leq \gamma_G(x, s) \leq \nu \gamma_\infty(x, s),$$

$$(4.19) \quad \lambda_2(x, s) - \ln(1 + \lambda_2(x, s)) \leq \gamma_F(x, s),$$

$$(4.20) \quad \lambda(x, s) \leq \lambda_2(x, s),$$

$$(4.21) \quad \gamma_G(x, s) \leq \lambda_2^2(x, s)(1 + \gamma_\infty(x, s)).$$

Moreover, if $\lambda_\infty(x, s) < 1$, then

$$(4.22) \quad \gamma_\infty(x, s) \leq \frac{\lambda_\infty(x, s)}{1 - \lambda_\infty(x, s)},$$

and, therefore,

$$(4.23) \quad \gamma_F(x, s) \leq \gamma_G(x, s) \leq \frac{\lambda_2^2(x, s)}{1 - \lambda_\infty(x, s)}.$$

Proof. The fact that $\lambda_\infty(x, s) \leq \lambda_2(x, s) \leq \nu^{1/2} \lambda_\infty(x, s)$ follows from Proposition 3.5. In order to prove the inequality $\lambda_\infty^+(x, s) \leq \lambda_\infty(x, s)$ we simply note that

$$\sigma_x \left(\frac{s}{\mu(x, s)} \right) = 1 + \sigma_x \left(\frac{s}{\mu(x, s)} + F'(x) \right).$$

Let us prove inequality (4.17). We have

$$\begin{aligned} \gamma_F(s, x) &= F(x) + F_*(s) + \nu \ln \mu(x, s) + \nu && \text{(in view of (4.8))} \\ &= -F_*(-F'(x)) - F(-F'_*(s)) + \nu \ln \mu(x, s) - \nu && \text{(using (2.13) and (2.14))} \\ &\leq \nu \ln \langle F'(x), F'_*(s) \rangle + \nu \ln \mu(x, s) - \nu \ln \nu && \text{(see (2.17))} \\ &= \nu \ln \left(1 + \frac{1}{\nu} \gamma_G(x, s) \right) && \text{(see (4.9))} \end{aligned}$$

and we get (4.17).

Next observe that, since all our proximity measures are homogeneous, we can assume for the rest of the proof that

$$(4.24) \quad \mu(x, s) = 1.$$

Thus, using (4.24) and (4.4)*, we get

$$\gamma_G(x, s) = \langle F'(x), F'_*(s) \rangle - \nu \leq \nu(\sigma_x(-F'_*(s)) - 1) = \nu \gamma_\infty(x, s),$$

which is exactly the right-hand inequality of (4.18). To prove the left-hand inequality, note that

$$\begin{aligned} \gamma_G(x, s) &= \|x + F'_*(s)\|_w^2 && \text{(by (4.15))} \\ &\geq \frac{\|x + F'_*(s)\|_x^2}{\sigma_x^2(w)} && \text{(by } C_* \text{ 4.1)} \\ &\geq \frac{|x + F'_*(s)|_x^2}{\sigma_x(-F'_*(s))} && \text{(by Lemma 3.4 and Proposition 3.5)} \\ &\geq \frac{|x + F'_*(s)|_x^2}{(1 + \gamma_\infty(x, s))} && \text{(by the definition of } \gamma_\infty(x, s)) \\ &\geq \frac{\sigma_x^2(-x - F'_*(s))}{(1 + \gamma_\infty(x, s))} \\ &= \frac{\gamma_\infty^2(x, s)}{(1 + \gamma_\infty(x, s))}. \end{aligned}$$

Note that the functional proximity measure is nonnegative in view of (2.17). Let us fix s and consider the function

$$\phi(v) := F(v) + F_*(s) + \langle s, v - x \rangle + \nu.$$

Note that $\phi(v) \geq \gamma_F(v, s)$ for any $v \in \text{int } K$ and $\phi(x) = \gamma_F(x, s)$. Note also that $\phi(v)$ is a strictly self-concordant function. Therefore, from P^* 2.2.2, we have

$$\phi(\bar{v}) \leq \phi(x) - [\lambda_\phi(x) - \ln(1 + \lambda_\phi(x))],$$

where

$$\bar{v} := x - \frac{1}{\lambda_\phi(x)} [\phi''(x)]^{-1} \phi'(x),$$

and $\lambda_\phi(x) := \langle \phi'(x), [\phi''(x)]^{-1} \phi'(x) \rangle^{1/2}$. Since $\phi(\bar{v}) \geq \gamma_F(\bar{v}, s) \geq 0$ and $\lambda_\phi(x) = \lambda_2(x, s)$, we get (4.19).

Further,

$$\lambda_2^2(x, s) = \langle s + F'(x), [F''(x)]^{-1}(s + F'(x)) \rangle = \langle s, [F''(x)]^{-1}s \rangle - \nu$$

(see (2.8) and (4.24)). Therefore, $\lambda(x, s)$ is well defined and we have

$$(4.25) \quad \lambda_2^2(x, s) = \frac{\nu \lambda^2(x, s)}{\nu - \lambda^2(x, s)} \geq \lambda^2(x, s)$$

which gives (4.20).

Let us now prove inequality (4.21). Let w be such that $s = F''(w)x$. In view of (4.15) and (4.24)

$$(4.26) \quad \gamma_G(x, s) = \langle s + F'(x), [F''(w)]^{-1}(s + F'(x)) \rangle.$$

Using inequality (4.7)* in C_* 4.1, we have

$$(4.27) \quad F''(w) \geq \frac{1}{\sigma_x(-F'_*(s))} F''(x).$$

Note that $\sigma_x(-F'_*(s)) = 1 + \gamma_\infty(x, s)$. Therefore, combining (4.26), (4.27), and (4.13) we come to (4.21).

Let $\lambda = \lambda_\infty(x, s) < 1$. This implies that for $p = s + F'(x)$ we have $-\lambda F'(x) + p \in K^*$. Hence,

$$s + (1 - \lambda)F'(x) \in K^*.$$

This means that

$$\frac{1}{\sigma_s(-F'(x))} \geq 1 - \lambda.$$

Therefore,

$$\gamma_\infty(x, s) = \sigma_s(-F'(x)) - 1 \leq \frac{\lambda}{1 - \lambda}$$

and inequality (4.22) is proved.

The remaining inequality (4.23) is a direct consequence of (4.17), (4.21), and (4.22). $\square$

From (2.17) we know that $\gamma_F(x, s)$ is nonnegative, and $\lambda(x, s)$ is nonnegative by definition. Also, if $\sigma := \sigma_x(s)$, then $-\sigma F'(x) - s \in K^*$, so $\nu \sigma - \langle s, x \rangle = \langle -\sigma F'(x) - s, x \rangle \geq 0$, which implies that $\lambda_\infty^+ = \sigma/\mu(x, s) - 1 \geq 0$. Hence, the theorem shows that all the measures are nonnegative. Now suppose that $s = -\mu(x, s)F'(x)$ (or equivalently that $x = -\mu(x, s)F'_*(s)$). Then $\gamma_\infty(x, s)$ and $\lambda_\infty(x, s)$ are zero, and hence all the measures are zero. Indeed, if any one of the measures is zero, we have $s = -\mu(x, s)F'(x)$ and hence all of the measures are zero. This follows from the conditions for equality in (2.17) for $\gamma_F(x, s)$, and hence for all the global measures. It

holds trivially for all the local measures except for $\lambda_\infty^+(x, s)$ and $\lambda(x, s)$. If $\lambda_\infty^+(x, s) = 0$, then as above $\sigma := \sigma_x(s) = \mu(x, s)$ and $\langle -\sigma F'(x) - s, x \rangle = 0$. But since $x \in \text{int } K$ and $-\sigma F'(x) - s \in K^*$, we find $s + \mu(x, s)F'(x) = s + \sigma F'(x) = 0$. If $\lambda(x, s)$ is zero, so is $\lambda_2(x, s)$ by (4.25).

Observe that (4.18) provides a very simple proof of T_* 5.2. Indeed, in our present notation, we need to show that (using Lemma 3.4)

$$(4.28) \quad \gamma_G(x, s) \geq \frac{3}{4}\mu(x, s)\sigma_x^2(w) - 1 = \frac{3}{4}\mu(x, s)\sigma_x(-F'_*(s)) - 1 = \frac{3}{4}\gamma_\infty(x, s) - \frac{1}{4}.$$

But this follows directly from (4.18) since $\gamma^2/(1 + \gamma) \geq (3\gamma - 1)/4$ for any $\gamma \geq 0$.

Note that all global proximity measures are unbounded: they tend to $+\infty$ when x or s approaches a nonzero point in the boundary of the corresponding cone. This is immediate for γ_F , and hence follows from the theorem for the other measures. All local proximity measures are bounded. Indeed, $v := [F''(x)]^{-1}s / \|s\|_x$ has $\|v\|_x = 1$, so $x - v \in K$ and $\langle s, x - v \rangle \geq 0$. This proves that $\langle s, x \rangle \geq \|s\|_x$ and hence, $0 \leq \lambda(x, s)^2 \leq \nu - 1$. This bounds $\lambda_2(x, s)$ using (4.25) and hence all the local proximity measures. It can also be shown that no contour of any local proximity measure defined by its maximum value corresponds to the boundary of the primal-dual cone; thus, no such measure can be transformed to a global proximity measure.

As we have seen, one of the key points in forming a primal-dual proximity measure is to decide what is the reference point on the central path. This decision results in the choice of the parameter μ involved in that measure. It is clear that any proximity measure leads to a specific rule for defining the optimal reference point. Indeed, we can consider a primal-dual proximity measure as a function of x , s , and μ . Therefore, the optimal choice for μ is given by minimizing the value of this measure with respect to μ when x and s are fixed. Let us demonstrate how our choice $\mu = \mu(x, s)$ arises in this way for the functional proximity measure.

Consider the penalty function

$$\psi_\mu(v) := \psi_\mu(x, y, s) = \frac{1}{\mu}[\langle c, x \rangle - \langle b, y \rangle] + F(x) + F_*(s).$$

By definition, for any $\mu > 0$ the point $v(\mu)$ is a minimizer of this function over the set

$$Q = \{v = (x, y, s) : Ax = b, A^*y + s = c, x \in K, y \in Y, s \in K^*\}.$$

Note that from Theorem 4.1 we know exactly what the minimal value of $\psi_\mu(v)$ over Q is:

$$\begin{aligned} \psi_\mu^* = \psi_\mu(v(\mu)) &= \frac{1}{\mu}[\langle c, x(\mu) \rangle - \langle b, y(\mu) \rangle] + F(x(\mu)) + F_*(s(\mu)) \\ &= \nu - \nu - \nu \ln \mu = -\nu \ln \mu. \end{aligned}$$

Thus, we can use the function

$$\gamma_\mu(v) = \psi_\mu(v) + \nu \ln \mu$$

as a kind of functional proximity measure.

Note that the function $\gamma_\mu(v)$ can estimate the distance from a given point v to the point $v(\mu)$ for any value of μ . But what is the point of the primal-dual central

path that is closest to v ? To answer this question we have to minimize $\gamma_\mu(v)$ as a function of μ . Since

$$\gamma_\mu(v) = \frac{1}{\mu}[\langle c, x \rangle - \langle b, y \rangle] + F(x) + F_*(s) + \nu \ln \mu,$$

we easily get that the optimal value of μ is

$$\mu^* = \frac{1}{\nu}[\langle c, x \rangle - \langle b, y \rangle] = \frac{1}{\nu}\langle s, x \rangle = \mu(x, s).$$

It is easy to see that $\gamma_{\mu(x,s)}(v) = \gamma_F(x, s)$.

The main advantage of the choice $\mu = \mu(x, s)$ is that the restriction of $\mu(x, s)$ to the feasible set is a linear function. But this choice is not optimal for other proximity measures. For example, for the local proximity measure λ_2 , let

$$(4.29) \quad \bar{\lambda}_2(x, s, \mu) := \left\| \frac{1}{\mu} s + F'(x) \right\|_x,$$

so that $\lambda_2(x, s) = \bar{\lambda}_2(x, s, \mu(x, s))$. Then the optimal choice of μ for $\bar{\lambda}_2$ is not $\mu(x, s)$ but

$$\bar{\mu} = \frac{\|s\|_x^2}{\langle s, x \rangle}.$$

This leads to the following relation:

$$(4.30) \quad \bar{\lambda}_2(x, s, \bar{\mu}) = \left[\nu - \frac{\langle s, x \rangle^2}{\|s\|_x^2} \right]^{1/2} \equiv \lambda(x, s).$$

The following lemma will be useful below.

LEMMA 4.3. *Let $x \in \text{int } K$, $s \in \text{int } K^*$, and $w \in \text{int } K$ be such that $F''(w)x = s$. Then for any $v \in E$, $u \in E^*$, we have*

$$\begin{aligned} \|v\|_x^2 &\leq \frac{1}{\mu(x, s)}[1 + \gamma_\infty(x, s)] \|v\|_w^2, \\ \|u\|_s^2 &\leq \frac{1}{\mu(x, s)}[1 + \gamma_\infty(x, s)] \|u\|_w^2. \end{aligned}$$

Proof. By (4.7)_{*} in C_* 4.1,

$$\frac{1}{\sigma_x(-F'_*(s))} F''(x) \leq F''(w),$$

and, similarly,

$$\frac{1}{\sigma_s(-F'(x))} F''_*(s) \leq [F''(w)]^{-1}.$$

The result follows since $\sigma_x(-F'_*(s)) = \sigma_s(-F'(x)) = [1 + \gamma_\infty(x, s)]/\mu(x, s)$ by definition of $\gamma_\infty(x, s)$. $\square$

5. Affine scaling and centering directions. In the next sections we will consider several primal-dual interior-point methods, which are based on two directions: the *affine-scaling* direction and the *centering* direction. Let us present the main properties of these directions.

5.1. The affine-scaling direction. We start from the definition. Let us fix points $x \in S^0(P)$ and $(y, s) \in S^0(D)$, and let $w \in \text{int } K$ be such that $s = F''(w)x$.

DEFINITION 5.1. *The affine-scaling direction for the primal-dual point (x, y, s) is the solution (p_x, p_y, p_s) of the following linear system:*

$$(5.1) \quad \begin{array}{rcl} F''(w)p_x & + & p_s = s, \\ & Ap_x & = 0, \\ A^*p_y & + & p_s = 0. \end{array}$$

Note that the first equation of (5.1) can be rewritten as follows:

$$(5.2) \quad p_x + [F''(w)]^{-1}p_s = x.$$

If $t := -F'(w) \in \text{int } K^*$, then $F''_*(t)s = x$ and $F''(t) = [F''(w)]^{-1}$, so the affine-scaling direction is defined symmetrically. It is also clear from the other two equations that

$$(5.3) \quad \langle p_s, p_x \rangle = 0.$$

For K the nonnegative orthant in R^n , this definition gives the usual primal-dual affine-scaling direction. For K the cone of positive semidefinite matrices of order n , efficient methods of computing w and the affine-scaling direction (and also the centering direction of the next subsection) are given in [TTT96].

The affine-scaling direction has several important properties.

LEMMA 5.2. *Let (p_x, p_y, p_s) be the affine-scaling direction for the strictly feasible point (x, y, s) . Then,*

$$(5.4) \quad \langle s, p_x \rangle + \langle p_s, x \rangle = \langle s, x \rangle,$$

$$(5.5) \quad \langle F'(x), p_x \rangle + \langle p_s, F'_*(s) \rangle = -\nu,$$

$$(5.6) \quad \|p_x\|_w^2 + \|p_s\|_w^2 = \langle s, x \rangle,$$

$$(5.7) \quad \bar{\sigma}(x, s) := \max\{|p_x|_x, |p_s|_s\} \geq 1.$$

Proof. Indeed, multiplying the first equation of (5.1) by x and using the definition of w we get

$$\langle s, x \rangle = \langle F''(w)p_x, x \rangle + \langle p_s, x \rangle = \langle s, p_x \rangle + \langle p_s, x \rangle,$$

which is (5.4). From this relation it is clear that the point $(x - p_x, y - p_y, s - p_s)$ is not strictly feasible (since $\langle s - p_s, x - p_x \rangle = 0$) and, therefore, we get (5.7).

Further, multiplying the first equation of (5.1) by $F'_*(s)$ and using the relation $F'(x) = F''(w)F'_*(s)$, we get (5.5) in view of (2.9).

Finally, multiplying the first equation of (5.1) by p_x we get

$$\|p_x\|_w^2 = \langle s, p_x \rangle$$

from (5.3). Similarly, multiplying (5.2) by p_s , we get

$$\|p_s\|_w^2 = \langle p_s, x \rangle.$$

Adding these equations we get (5.6) in view of (5.4). $\square$

It is interesting that the affine-scaling direction is related in a very nice way with the functional proximity measure.

THEOREM 5.3. *Let the point (x, y, s) be strictly feasible, and let $w \in \text{int } K$ be such that $F''(w)x = s$. Then for small enough $\alpha \geq 0$ the point*

$$x(\alpha) := x - \alpha p_x, \quad y(\alpha) := y - \alpha p_y, \quad s(\alpha) := s - \alpha p_s$$

is also strictly feasible and for any such α

$$\begin{aligned} \gamma_F(x(\alpha), s(\alpha)) - \gamma_F(x, s) &= F(x - \alpha p_x) + F\left(x + \frac{\alpha}{1 - \alpha} p_x\right) - 2F(x) \\ &= F\left(x - \frac{\alpha^2}{2(1 - \alpha)} [F''(x)]^{-1} F'''(x) [p_x, [F''(w)]^{-1} p_s]\right) - F(x). \end{aligned} \quad (5.8)$$

Proof. Since the point (x, y, s) is strictly feasible, for small enough $\alpha \geq 0$ the point $(x(\alpha), y(\alpha), s(\alpha))$ is strictly feasible by the second and third equations of (5.1).

In view of (5.4) and (5.3), we get the following relation for the duality gap:

$$(5.9) \quad \langle s(\alpha), x(\alpha) \rangle = \langle s, x \rangle - \alpha [\langle s, p_x \rangle + \langle p_s, x \rangle] = (1 - \alpha) \langle s, x \rangle.$$

Therefore,

$$(5.10) \quad \nu \ln \langle s(\alpha), x(\alpha) \rangle - \nu \ln \langle s, x \rangle = \nu \ln(1 - \alpha).$$

Further, from (5.1), we have

$$\begin{aligned} s(\alpha) &= s - \alpha p_s \\ &= s - \alpha [s - F''(w)p_x] \\ &= (1 - \alpha)s + \alpha F''(w)p_x \\ &= F''(w)[(1 - \alpha)x + \alpha p_x]. \end{aligned}$$

Therefore, using (2.19), we obtain

$$(5.11) \quad \begin{aligned} F_*(s(\alpha)) &= F((1 - \alpha)x + \alpha p_x) - 2F(w) - \nu \\ &= F((1 - \alpha)x + \alpha p_x) + F_*(s) - F(x). \end{aligned}$$

Thus, in view of (5.10) and (5.11), we get

$$\begin{aligned} \gamma_F(x(\alpha), s(\alpha)) - \gamma_F(x, s) &= \nu \ln(1 - \alpha) + F(x(\alpha)) + F_*(s(\alpha)) - F(x) - F_*(s) \\ &= \nu \ln(1 - \alpha) + F(x(\alpha)) + F((1 - \alpha)x + \alpha p_x) - 2F(x), \end{aligned}$$

which is the first part of (5.8) by (2.6).

Further, let

$$v = x - \frac{\alpha^2}{2(1 - \alpha)} [F''(x)]^{-1} F'''(x) [p_x, [F''(w)]^{-1} p_s], \quad \beta = \frac{\alpha}{1 - \alpha}.$$

Then,

$$\beta - \alpha = \beta\alpha = \frac{\alpha^2}{1 - \alpha}$$

and we have

$$\begin{aligned} v &= x - \frac{1}{2} \beta\alpha [F''(x)]^{-1} F'''(x) [p_x, (p_x + [F''(w)]^{-1} p_s) - p_x] \\ &= x - \frac{1}{2} \beta\alpha [F''(x)]^{-1} F'''(x) [p_x, x - p_x] \\ &= x + (\beta - \alpha)p_x + \frac{1}{2} \beta\alpha [F''(x)]^{-1} F'''(x) [p_x, p_x]. \end{aligned}$$

Therefore, the second part of (5.8) follows from Theorem 3.11. $\square$

Denote

$$p(x, s) = -\frac{1}{2}[F''(x)]^{-1}F'''(x)[p_x, [F''(w)]^{-1}p_s].$$

In view of the above theorem we need more information about this direction.

LEMMA 5.4.

$$(5.12) \quad \langle F'(x), p(x, s) \rangle \leq 2(1 + \lambda_\infty^+(x, s))\sqrt{\gamma_G(x, s)} \|p_x\|_x \cdot \|p_s\|_s,$$

$$(5.13) \quad \langle F'(x), p(x, s) \rangle \leq \nu(1 + \gamma_\infty(x, s))\sqrt{\gamma_F(x, s)},$$

$$(5.14) \quad \|p(x, s)\|_x \leq \|p_x\|_x \|p_s\|_s \leq \frac{1}{2}\nu(1 + \gamma_\infty(x, s)).$$

Proof. Indeed,

$$\begin{aligned} \langle F'(x), p(x, s) \rangle &= -\frac{1}{2}\langle F'''(x)[p_x, [F''(w)]^{-1}p_s], [F''(x)]^{-1}F'(x) \rangle \\ &= \frac{1}{2}F'''(x)[x, p_x, [F''(w)]^{-1}p_s] \\ &= -\langle F''(x)p_x, [F''(w)]^{-1}p_s \rangle \\ &= \left\langle \left[\frac{1}{\mu(x, s)}F''(w) - F''(x) \right] p_x, [F''(w)]^{-1}p_s \right\rangle. \end{aligned}$$

Also note that

$$\|[F''(w)]^{-1}p_s\|_x = \|p_s\|_s.$$

Therefore, using Theorem 3.8 we obtain

$$(5.15) \quad \langle F'(x), p(x, s) \rangle \leq \frac{2}{(1 - \sigma_x(p))^2} \|p\|_x \cdot \|p_x\|_x \cdot \|p_s\|_s,$$

where $p = x - \sqrt{\mu(x, s)}w$. In view of (3.19) and (3.21) we have

$$(5.16) \quad \|p\|_x \leq \sqrt{\gamma_G(x, s)}.$$

Further, by definition $\sigma_x(p) = \frac{1}{\alpha}$, where α is the maximal step such that

$$x - \alpha(x - \sqrt{\mu(x, s)}w) \in K.$$

Clearly, such α is greater than one. Therefore,

$$\frac{\alpha - 1}{\alpha\sqrt{\mu(x, s)}} = \frac{1}{\sigma_w(x)}.$$

Thus,

$$\frac{1}{(1 - \sigma_x(p))^2} = \frac{\alpha^2}{(\alpha - 1)^2} = \frac{\sigma_w(x)^2}{\mu(x, s)} = \frac{\sigma_s(x)}{\mu(x, s)} = \lambda_\infty^+(x, s) + 1$$

(see Lemma 3.4). Combining this equality with (5.15) and (5.16) we get (5.12).

In order to prove (5.13), note that in view of Theorem 3.8,

$$\begin{aligned} & \left| \left\langle \left[\frac{1}{\mu(x, s)} F''(w) - F''(x) \right] p_x, [F''(w)]^{-1} p_s \right\rangle \right| \\ &= \frac{1}{\mu(x, s)} \left| \left\langle \left[F''(x/\sqrt{\mu(x, s)}) - F''(w) \right] p_x, [F''(w)]^{-1} p_s \right\rangle \right| \\ &\leq \frac{2}{\mu(x, s)(1 - \sigma_w(p))^2} \|p\|_w \cdot \|p_x\|_w \cdot \|p_s\|_w, \end{aligned}$$

where now $p = w - x/\sqrt{\mu(x, s)}$. Note that $\|p\|_w^2 \leq \gamma_F(x, s)$ by Lemma 3.12,

$$2 \|p_x\|_w \cdot \|p_s\|_w \leq \|p_x\|_w^2 + \|p_s\|_w^2 = \nu\mu(x, s)$$

in view of (5.6), and

$$\sigma_w(p) = 1 - \sqrt{\frac{1}{1 + \gamma_\infty(x, s)}}$$

(the proof is quite straightforward). Combining these inequalities, we get (5.13).

Finally, the first inequality in (5.14) follows from Theorem 3.8. In order to prove the second one, note that from Lemma 4.3 and (5.6) we have

$$\begin{aligned} 2 \|p_x\|_x \cdot \|p_s\|_s &\leq \|p_x\|_x^2 + \|p_s\|_s^2 \\ &\leq \frac{1}{\mu(x, s)} (1 + \gamma_\infty(x, s)) (\|p_x\|_w^2 + \|p_s\|_w^2) \\ &= \nu(1 + \gamma_\infty(x, s)). \quad \square \end{aligned}$$

5.2. Centering direction. Let us again fix points $x \in S^0(P)$ and $(y, s) \in S^0(D)$, and let $w \in \text{int } K$ be such that $s = F''(w)x$.

DEFINITION 5.5. *The centering direction for the primal-dual point (x, y, s) is the solution (d_x, d_y, d_s) of the following linear system:*

$$(5.17) \quad \begin{array}{rcl} F''(w)d_x & + & d_s = s + \mu(x, s)F'(x), \\ Ad_x & & = 0, \\ A^*d_y & + & d_s = 0. \end{array}$$

Note that the first equation of (5.17) can be rewritten as follows:

$$(5.18) \quad d_x + [F''(w)]^{-1}d_s = x + \mu(x, s)F'_*(s),$$

so the centering direction is also symmetric between the primal and dual. It is also clear from the other two equations that

$$(5.19) \quad \langle d_s, d_x \rangle = 0.$$

Let us present the main properties of the centering direction.

LEMMA 5.6. *Let (d_x, d_y, d_s) be the centering direction for the strictly feasible point (x, y, s) . Then,*

$$(5.20) \quad \langle s, d_x \rangle + \langle d_s, x \rangle = 0,$$

$$(5.21) \quad \langle F'(x), d_x \rangle + \langle d_s, F'_*(s) \rangle = \gamma_G(x, s),$$

$$(5.22) \quad \|d_x\|_w^2 + \|d_s\|_w^2 = \mu(x, s)\gamma_G(x, s),$$

$$(5.23) \quad \|d_x\|_x^2 + \|d_s\|_s^2 \leq \gamma_G(x, s)(1 + \gamma_\infty(x, s)).$$

Moreover, the affine-scaling and the centering directions are orthogonal in the metric defined by w :

$$(5.24) \quad \langle F''(w)p_x, d_x \rangle + \langle d_s, [F''(w)]^{-1}p_s \rangle = 0.$$

Proof. Indeed, multiplying the first equation of (5.17) by x , we get (5.20) in view of (2.9). Multiplying the first equation of (5.17) by $F'_*(s)$ and using again (2.9), we obtain

$$\langle F'(x), d_x \rangle + \langle d_s, F'_*(s) \rangle = -\nu + \mu(x, s)\langle F'(x), F'_*(s) \rangle = \gamma_G(x, s).$$

Further, multiplying the first equation of (5.17) by d_x and using (5.19) we get

$$\|d_x\|_w^2 = \langle s, d_x \rangle + \mu(x, s)\langle F'(x), d_x \rangle.$$

Similarly, from (5.18) we get

$$\|d_s\|_w^2 = \langle d_s, x \rangle + \mu(x, s)\langle d_s, F'_*(s) \rangle.$$

Adding these equations, we get (5.22) from (5.20) and (5.21).

Next, (5.23) follows from (5.22) and Lemma 4.3.

In order to prove (5.24), let us note that $\langle p_s, d_x \rangle = \langle d_s, p_x \rangle = 0$. Therefore,

$$\begin{aligned} \langle F''(w)p_x, d_x \rangle + \langle d_s, [F''(w)]^{-1}p_s \rangle &= \langle s - p_s, d_x \rangle + \langle d_s, x - p_x \rangle \\ &= \langle s, d_x \rangle + \langle d_s, x \rangle = 0 \end{aligned}$$

in view of (5.20). $\square$

Note that (5.20) shows that the centering direction keeps the duality gap constant.

Let us describe the behavior of the Newton process based on the centering direction. Let the point (x, y, s) be strictly feasible. Consider the following process:

$$(5.25) \quad \left. \begin{aligned} x_0 &= x, \quad y_0 = y, \quad s_0 = s; \\ x_{k+1} &= x_k - \alpha_k d_{x_k}, \\ y_{k+1} &= y_k - \alpha_k d_{y_k}, \\ s_{k+1} &= s_k - \alpha_k d_{s_k}, \end{aligned} \right\} k = 0, \dots,$$

where

$$(5.26) \quad \alpha_k = \frac{1}{1 + \gamma_\infty(x_k, s_k) + \bar{\sigma}_k}, \quad \bar{\sigma}_k = \max\{\sigma_{x_k}(d_{x_k}), \sigma_{s_k}(d_{s_k})\}.$$

THEOREM 5.7. *For any $k \geq 0$ we have*

$$(5.27) \quad \gamma_F(x_{k+1}, s_{k+1}) \leq \gamma_F(x_k, s_k) - [\tau_k - \ln(1 + \tau_k)],$$

where

$$\tau_k = \left(\frac{\gamma_G(x_k, s_k)}{1 + \gamma_\infty(x_k, s_k)} \right)^{1/2} \geq \frac{\gamma_\infty(x_k, s_k)}{1 + \gamma_\infty(x_k, s_k)}.$$

Proof. In order to simplify the notation let us omit subscripts for the current iterate and use a subscript $+$ for the next iterate.

Since from Lemma 5.6 the centering direction does not change the duality gap, the decrease of the functional proximity measure can be estimated as follows, using T_* 4.2 and P_* 4.1:

$$\begin{aligned}
\gamma_F(x_+, s_+) - \gamma_F(x, s) &= F(x_+) - F(x) + F_*(s_+) - F_*(s) \\
&\leq -\alpha \langle F'(x), d_x \rangle + \left(\frac{\|d_x\|_x}{\sigma_x(d_x)} \right)^2 (-\alpha \sigma_x(d_x) - \ln(1 - \alpha \sigma_x(d_x))) \\
&\quad -\alpha \langle F'_*(s), d_s \rangle + \left(\frac{\|d_s\|_s}{\sigma_s(d_s)} \right)^2 (-\alpha \sigma_s(d_s) - \ln(1 - \alpha \sigma_s(d_s))) \\
&\leq -\alpha \gamma_G(x, s) + \frac{1}{\bar{\sigma}^2} (\|d_x\|_x^2 + \|d_s\|_s^2) (-\alpha \bar{\sigma} - \ln(1 - \alpha \bar{\sigma})) \\
&\leq -\gamma_G(x, s) \left[\alpha + \frac{1 + \gamma_\infty(x, s)}{\bar{\sigma}^2} (\alpha \bar{\sigma} + \ln(1 - \alpha \bar{\sigma})) \right].
\end{aligned}$$

Maximizing the expression in brackets with respect to α , we get the following step size yielding the optimal bound:

$$\alpha = \frac{1}{1 + \gamma_\infty(x, s) + \bar{\sigma}}.$$

Substituting this value in the above chain of inequalities, we obtain the following:

$$\begin{aligned}
\gamma_F(x_+, s_+) - \gamma_F(x, s) &\leq -\frac{1}{\bar{\sigma}^2} \gamma_G(x, s) (1 + \gamma_\infty(x, s)) \left[\frac{\bar{\sigma}}{1 + \gamma_\infty(x, s)} \right. \\
&\quad \left. - \ln \left(1 + \frac{\bar{\sigma}}{1 + \gamma_\infty(x, s)} \right) \right].
\end{aligned}$$

Note that the expression in brackets above is monotonically decreasing in $\bar{\sigma}$ and that

$$\bar{\sigma}^2 \leq \|d_x\|_x^2 + \|d_s\|_s^2 \leq (1 + \gamma_\infty(x, s)) \gamma_G(x, s).$$

Substituting this bound in the previous expression, we get the desired inequality for the change in γ_F . The bound on τ_k follows from (4.18). $\square$

6. Some closely path-following methods. In this section we describe some extensions of standard primal-dual path-following methods for linear programming to the setting of self-scaled cones. All our algorithms generate iterates in the (narrow) neighborhood

$$(6.1) \quad \mathcal{N}(\beta) := \left\{ (x, y, s) \in S^0(P) \times S^0(D) : \lambda_2(x, s) = \left\| \frac{s}{\mu(x, s)} + F'(x) \right\|_x \leq \beta \right\}$$

of the primal-dual central path, where $\beta \in (0, 1)$. Note that this definition is symmetric between x and s (see (4.13)).

Suppose we are given, at the start of some iteration, a point $(x, y, s) \in \mathcal{N}(\beta)$ with

$$(6.2) \quad \delta := \left\| \frac{s}{\mu(x, s)} + F'(x) \right\|_x = \lambda_\infty(x, s) \leq \epsilon := \left\| \frac{s}{\mu(x, s)} + F'(x) \right\|_x = \lambda_2(x, s) \leq \beta.$$

Actually, we could work with the slightly wider neighborhood $\mathcal{N}'(\beta) := \{(x, y, s) \in S^0(P) \times S^0(D) : \lambda(x, s) \leq \beta\}$ for $\beta \in (0, 1)$. Note that $(x, y, s) \in S^0(P) \times S^0(D)$ lies in this neighborhood if $\bar{\lambda}_2(x, s, \mu) = \|s/\mu + F'(x)\|_x \leq \beta$ for some $\mu > 0$ by (4.30).

We could then try to decrease the value of μ and find new iterates for which this inequality remains true. However, it is easy to see that the inequality implies that μ is very close to $\mu(x, s)$.

LEMMA 6.1. *If $\|s/\mu + F'(x)\|_x \leq \beta$ holds for some $\mu > 0$, then*

$$(6.3) \quad \left(1 - \frac{\beta}{\sqrt{\nu}}\right) \mu \leq \mu(x, s) \leq \left(1 + \frac{\beta}{\sqrt{\nu}}\right) \mu.$$

Proof. To prove the upper bound, note that

$$\begin{aligned} \langle s, x \rangle &= \mu \left(\left\langle \frac{s}{\mu} + F'(x), x \right\rangle - \langle F'(x), x \rangle \right) \\ &\leq \mu \left(\left\| \frac{s}{\mu} + F'(x) \right\|_x \|x\|_x + \nu \right) \\ &\leq (\nu + \beta\sqrt{\nu})\mu, \end{aligned}$$

where we have used (2.9) and (2.10). Now divide by ν . The lower bound is proved similarly. $\square$

If we can find, for every such $(x, y, s) \in \mathcal{N}(\beta)$, some $(x_+, y_+, s_+) \in \mathcal{N}(\beta)$ with $\mu(x_+, s_+) \leq (1 - \kappa/\sqrt{\nu})\mu(x, s)$, for some positive constant κ , then clearly we can construct an algorithm which, given $(x_0, y_0, s_0) \in \mathcal{N}(\beta)$, achieves $(x_k, y_k, s_k) \in \mathcal{N}(\beta)$ with $\langle s_k, x_k \rangle \leq \epsilon$ within $O(\sqrt{\nu} \ln(\langle s_0, x_0 \rangle / \epsilon))$ iterations. All the algorithms of this section are of this type.

Now let us denote

$$\mu := \mu(x, s)$$

and examine the effect on (6.2) if we replace μ by

$$(6.4) \quad \mu_+ := \left(1 - \frac{\kappa}{\sqrt{\nu}}\right) \mu$$

for some fixed constant κ .

LEMMA 6.2. *Suppose (6.2) holds and μ_+ is given by (6.4). Then,*

$$(6.5) \quad \delta_+ := \left\| \frac{s}{\mu_+} + F'(x) \right\|_x \leq \epsilon_+ := \left\| \frac{s}{\mu_+} + F'(x) \right\|_x$$

satisfy

$$(6.6) \quad \delta_+ \leq \frac{\delta + \kappa}{1 - \kappa}, \quad \epsilon_+ \leq \frac{\epsilon + \kappa}{1 - \kappa}.$$

Proof. Using (2.10), we have

$$\begin{aligned} \epsilon_+ &= \left\| \frac{s}{\mu_+} + F'(x) \right\|_x \leq \frac{\mu}{\mu_+} \left[\left\| \frac{s}{\mu} + F'(x) \right\|_x + \left(1 - \frac{\mu_+}{\mu}\right) \|F'(x)\|_x \right] \\ &\leq \frac{1}{1 - \kappa} \left[\epsilon + \frac{\kappa}{\sqrt{\nu}} \sqrt{\nu} \right] \end{aligned}$$

as desired. We proceed similarly for δ_+ , using $\|F'(x)\|_x \leq \|F'(x)\|_x$. $\square$

Having decreased μ , we now need to update x , y , and s so that (6.2) remains true at the new iterate. Hence, with $w \in \text{int } K$ such that $F''(w)x = s$, let q_x , q_s , and q_y solve

$$(6.7) \quad \begin{array}{rcl} F''(w)q_x & + & q_s = s + \mu_+ F'(x), \\ Aq_x & & = 0, \\ A^*q_y & + & q_s = 0. \end{array}$$

Note that the first equation can alternatively be written as $q_x + F''_*(t)q_s = x + \mu_+ F'_*(s)$, where $t := -F'(w) \in \text{int } K^*$ satisfies $F''_*(t)s = x$, so that this is a symmetric system. Also note that (q_x, q_y, q_s) can be written in terms of the affine-scaling and centering directions of section 5. From Definitions 5.1 and 5.5 we have

$$(6.8) \quad (q_x, q_y, q_s) = \frac{\mu - \mu_+}{\mu} (p_x, p_y, p_s) + \frac{\mu_+}{\mu} (d_x, d_y, d_s).$$

Now set

$$(6.9) \quad x_+ := x - q_x, \quad y_+ := y - q_y, \quad s_+ := s - q_s.$$

Clearly, $Ax_+ = b$ and $A^*y_+ + s_+ = c$. Also, since $\langle q_s, q_x \rangle = 0$, we have

$$\begin{aligned} \langle s_+, x_+ \rangle &= \langle s, x \rangle - \langle q_s, x \rangle - \langle s, q_x \rangle \\ &= \langle s, x \rangle - \langle F''(w)q_x + q_s, x \rangle \\ &= \langle s, x \rangle - \langle s + \mu_+ F'(x), x \rangle \\ &= \nu \mu_+. \end{aligned}$$

This also follows from (6.8) and Lemmas 5.2 and 5.6. Hence,

$$(6.10) \quad \mu_+ = \mu(x_+, s_+).$$

LEMMA 6.3. *Suppose*

$$(6.11) \quad \eta := \frac{\epsilon_+}{1 - \delta}$$

satisfies $\eta < 1$ (so that $\delta_+ \leq \epsilon_+ < 1$). Then,

$$\begin{aligned} \|q_x\|_x &\leq \eta, \quad \|q_s\|_s \leq \eta, \\ x_+ &\in S^0(P), \quad (y_+, s_+) \in S^0(D), \end{aligned}$$

and for any $v \in E$, $u \in E^*$,

$$(6.12) \quad \|v\|_{x_+} \leq \frac{1}{1 - \eta} \|v\|_x, \quad \|u\|_{x_+} \leq (1 + \eta) \|u\|_x.$$

Proof. We have

$$\begin{aligned} \|q_x\|_x^2 &\leq [\mu(1 - \delta)]^{-1} \|q_x\|_w^2 \quad (\text{by Lemma 3.13}) \\ &= [\mu(1 - \delta)]^{-1} \langle F''(w)q_x + q_s, q_x \rangle \\ &= [\mu(1 - \delta)]^{-1} \langle s + \mu_+ F'(x), q_x \rangle \\ &\leq \left(\frac{\mu_+}{\mu} \right) (1 - \delta)^{-1} \left\| \frac{s}{\mu_+} + F'(x) \right\|_x \|q_x\|_x \\ &\leq \eta \|q_x\|_x, \end{aligned}$$

which gives the first inequality, and similarly $\|q_s\|_s \leq \eta$. If $\eta < 1$, $x - q_x \in \text{int } K$, so $x_+ \in S^0(P)$, and similarly $(y_+, s_+) \in S^0(D)$. Then (6.12) follows from Theorem 3.6. $\square$

Note that it does not seem to be easy to get a good bound (depending on δ 's, not ϵ 's) on $\|q_x\|_x$ or $\|q_s\|_s$.

We can now assemble these results to get the following theorem.

THEOREM 6.4. *Let μ_+ , x_+ , y_+ , s_+ , δ_+ , ϵ_+ , and η be defined as above, and let $\eta < 1$. Then,*

$$(6.13) \quad \lambda_2(x_+, s_+) = \bar{\lambda}_2(x_+, s_+, \mu_+) = \left\| \frac{s_+}{\mu_+} + F'(x_+) \right\|_{x_+} \leq \frac{(2 + \delta - \kappa)(\epsilon + \kappa)^2}{(1 - \delta)^2(1 - \kappa)^3}.$$

Proof. The two equations follow from the definitions and (6.10). For the inequality, we write

$$\begin{aligned} s_+ + \mu_+ F'(x_+) &= s - q_s + \mu_+ F'(x - q_x) \\ &= s - q_s + \mu_+ F'(x) + \mu_+ (F'(x - q_x) - F'(x)) \\ &= [s + \mu_+ F'(x) - (q_s + F''(w)q_x)] + [(F''(w) - \mu_+ F''(x))q_x] \\ &\quad + \mu_+ [F'(x - q_x) - F'(x) - F''(x)(-q_x)]. \end{aligned}$$

The first term vanishes by the definition of our steps. Hence,

$$\begin{aligned} \left\| \frac{s_+}{\mu_+} + F'(x_+) \right\|_{x_+} &\leq \left\| \left[\frac{F''(w)}{\mu_+} - F''(x) \right] q_x \right\|_{x_+} \\ &\quad + \|F'(x - q_x) - F'(x) - F''(x)(-q_x)\|_{x_+}. \end{aligned}$$

Now applying Lemma 6.3 and Corollary 3.14 (with μ_+ and δ_+ replacing μ and δ) to the first term, we find

$$\begin{aligned} \left\| \left[\frac{F''(w)}{\mu_+} - F''(x) \right] q_x \right\|_{x_+} &\leq (1 + \eta) \left\| \left[\frac{F''(w)}{\mu_+} - F''(x) \right] q_x \right\|_x \\ &\leq (1 + \eta)\delta_+ \|q_x\|_x \leq (1 + \delta_+)\eta^2. \end{aligned}$$

Also, T_* 4.3 bounds the second term by $\|q_x\|_x \|q_x\|_x \leq \eta^2$, so we conclude that

$$(6.14) \quad \left\| \frac{s_+}{\mu_+} + F'(x_+) \right\|_{x_+} \leq (2 + \delta_+)\eta^2 = \frac{2 + \delta_+}{(1 - \delta)^2} \epsilon_+^2.$$

Substituting the upper bounds on δ_+ and ϵ_+ from Lemma 6.2 yields the conclusion of the theorem. $\square$

We can now analyze some algorithms. First, suppose we choose $\beta = 1/10$ and $\kappa = 1/15$. Then $\delta \leq \epsilon \leq 1/10$, $\delta_+ \leq \epsilon_+ \leq 5/28$, $\eta \leq 50/252 < 1$, and $(2 + \delta - \kappa)(\epsilon + \kappa)^2 / [(1 - \delta)^2(1 - \kappa)^3] < 1/10$. Thus these choices ensure that all iterates lie in $\mathcal{N}(1/10)$ and that μ decreases by a factor of $[1 - (15\sqrt{\nu})^{-1}]$ at each iteration. In summary, we have the following method.

ALGORITHM 6.1 (short-step method). *Let $\beta = 1/10$ and $\kappa = 1/15$, and choose $\zeta > 0$. Given $(x_0, s_0) \in \mathcal{N}(\beta)$, set $k = 0$;*

While $\langle s_k, x_k \rangle > \zeta$ *do*
begin
 set $(x, y, s) := (x_k, y_k, s_k)$ *and* $\mu := \mu(x, s)$;
 compute μ_+ *from* (6.4);
 compute the directions (q_x, q_y, q_s) *from* (6.7);
 set $(x_{k+1}, y_{k+1}, s_{k+1}) := (x_+, y_+, s_+)$ *from* (6.9);
 $k := k + 1$;
end.

Thus we have extended the short-step algorithm of Monteiro and Adler [MA89] and Kojima, Mizuno, and Yoshise [KMY89] to the setting of self-scaled cones.

If our strategy is, as above, to first decrease μ to μ_+ so that ϵ_+ remains bounded, and then to take one or more steps to get close again with the same value of μ , then we cannot achieve much more than the reduction in μ given above. Indeed, Lemma 6.1 shows that if $\bar{\lambda}_2(x, s, \mu_i) \leq \beta$ for two values μ_1 and μ_2 , then we must have $\mu_2 \geq (1 - 2\beta/\sqrt{\nu})\mu_1$.

However, better progress is possible using adaptive steps. First, suppose we calculate the steps q_x , q_y , and q_s parametrically in μ_+ and choose the *smallest* μ_+ so that (x_+, y_+, s_+) remains in $\mathcal{N}(1/10)$. By the analysis above, we can choose $\mu_+ \leq [1 - (15\sqrt{\nu})^{-1}]\mu$ at each iteration, and thus achieve the same complexity estimate as for the short-step method.

ALGORITHM 6.2 (adaptive-step method I). *Let* $\beta = 1/10$, *and choose* $\zeta > 0$. *Given* $(x_0, s_0) \in \mathcal{N}(\beta)$, *set* $k = 0$;

While $\langle s_k, x_k \rangle > \zeta$ *do*
begin
 set $(x, y, s) := (x_k, y_k, s_k)$ *and* $\mu := \mu(x, s)$;
 compute the directions (q_x, q_y, q_s) *parametrically in* μ_+ *from* (6.7);
 choose μ_+ *as small as possible so that* (x_+, y_+, s_+) *defined by* (6.9) *lies in* $\mathcal{N}(\beta)$;
 set $(x_{k+1}, y_{k+1}, s_{k+1}) := (x_+, y_+, s_+)$;
 $k := k + 1$;
end.

This algorithm extends the “largest-step path-following” algorithm which was suggested by Monteiro and Adler and made precise and implemented by Mizuno, Yoshise, and Kikuchi [MYK89].

The second adaptive-step algorithm we consider, and the last in this section, is an extension of the predictor-corrector method of Mizuno, Todd, and Ye [MTY90]. In this method, each step consists of two substeps. In the first, we strive to reduce the duality gap while not straying too far from the central path (as measured by the proximity measure λ_2) by taking a step along the affine-scaling direction, while in the second, we take a single step in the centering direction so that the duality gap remains constant and the iterates return to close proximity with the path. In the next section, we will consider a much more sophisticated version of this strategy, where we use the global proximity measure γ_F so that much longer steps can be taken, and correct by a sequence of centering steps as in section 5.2.

We first consider the predictor step. Here the direction is (p_x, p_y, p_s) from Definition 5.1; so it satisfies (6.7) with $\mu_+ = 0$. We still assume that our iterate (x, y, s) satisfies (6.2). As in Lemma 6.3 (see also the proof of Lemma 5.2), we find

$$\begin{aligned}
 \|p_x\|_x^2 &\leq [\mu(1 - \delta)]^{-1} \|p_x\|_w^2 \\
 &= [\mu(1 - \delta)]^{-1} \langle F'''(w)p_x + p_s, p_x \rangle
 \end{aligned}$$

$$\begin{aligned}
&= [\mu(1-\delta)]^{-1} \langle s, p_x \rangle \\
&\leq [\mu(1-\delta)]^{-1} \|s\|_x \|p_x\|_x,
\end{aligned}$$

so

$$\begin{aligned}
\|p_x\|_x &\leq (1-\delta)^{-1} \left\| \frac{s}{\mu} + F'(x) - F'(x) \right\|_x \\
&\leq (1-\delta)^{-1} (\epsilon + \sqrt{\nu}) \leq \frac{1+\epsilon}{1-\delta} \sqrt{\nu}.
\end{aligned}$$

We choose the step size α as large as possible so that the next iterates remain approximately centered as measured by λ_2 . We will show that, for a certain value of the constant κ ,

$$\alpha := \frac{\kappa}{\sqrt{\nu}}$$

suffices. Note that then

$$\|\alpha p_x\|_x \leq \frac{1+\epsilon}{1-\delta} \kappa =: \eta,$$

and we choose κ so that $\eta < 1$. Similarly, $\|\alpha p_s\|_s \leq \eta$. So if we set

$$(6.15) \quad x^+ := x - \alpha p_x, \quad y^+ := y - \alpha p_y, \quad s^+ := s - \alpha p_s,$$

we will have $x^+ \in S^0(P)$ and $(y^+, s^+) \in S^0(D)$. Let us continue to assume that α is given as above. Note that from (5.9)

$$\langle s^+, x^+ \rangle = (1-\alpha) \langle s, x \rangle.$$

So we set $\mu^+ := (1-\alpha)\mu = \mu(x^+, s^+)$. Then we find

$$\begin{aligned}
\lambda_2(x^+, s^+) &= \left\| \frac{s^+}{\mu^+} + F'(x^+) \right\|_{x^+} \\
&= \left\| \frac{(s - \alpha p_s)}{(1-\alpha)\mu} + F'(x - \alpha p_x) \right\|_{x^+} \\
&= \left\| \frac{[(1-\alpha)s + \alpha(s - p_s)]}{[(1-\alpha)\mu]} + F'(x - \alpha p_x) \right\|_{x^+} \\
&\leq \left\| \frac{s}{\mu} + F'(x) \right\|_{x^+} + \|F'(x - \alpha p_x) - F'(x) - F''(x)(-\alpha p_x)\|_{x^+} \\
&\quad + \left\| \alpha \left[\frac{F''(w)}{(1-\alpha)\mu} - F''(x) \right] p_x \right\|_{x^+}.
\end{aligned}$$

The first term above is at most $(1+\eta)\epsilon$ using Theorem 3.6. By T_* 4.3, the second term is bounded by η^2 . Now (3.22) gives two-sided bounds for $F''(w)/\mu$. Dividing by

$(1 - \alpha)$, we see that the inequalities remain true with μ replaced with $(1 - \alpha)\mu$ and δ replaced with

$$\delta' := \frac{\delta + \alpha}{1 - \alpha}.$$

Then, following the argument in the proof of Corollary 3.14, we discover that the third term is bounded by $(1 + \eta)\delta'\eta$. Using our form of α , we conclude that

$$\left\| \frac{s^+}{\mu^+} + F'(x^+) \right\|_{x^+} \leq (1 + \eta)\epsilon + \eta^2 + \frac{\epsilon + \kappa}{1 - \kappa}(\eta + \eta^2).$$

Note that if $\epsilon \leq 1/10$ and we choose $\kappa := 1/10$, then $\eta \leq 1/8$ and the bound above is less than $1/6$. Thus we have proved the following theorem.

THEOREM 6.5. *If (x, y, s) satisfies (6.2) with $\beta = 1/10$ and if we choose as our step size α^* the largest α so that the point defined by (6.15) lies in $\mathcal{N}(1/6)$, then $\alpha^* \geq (10\sqrt{\nu})^{-1}$, and*

$$\mu^+ = [1 - \alpha^*]\mu \leq [1 - (10\sqrt{\nu})^{-1}]\mu. \quad \square$$

We next apply a corrector step to the resulting iterate (x^+, y^+, s^+) to return to $\mathcal{N}(1/10)$. For this, note that we now have a point satisfying (6.2) with $\beta = 1/6$. We proceed as in the first part of this section, but choosing $\kappa = 0$, so that we take a unit step in the centering direction (d_x, d_y, d_s) of Definition 5.5 (so that we apply one step of the centering process (5.25) with a unit step). We find that the corresponding value of η is at most $1/5 < 1$, and, from Theorem 6.4, that the result (x_+, y_+, s_+) of the corrector step lies in $\mathcal{N}(1/10)$ with $\mu_+ := \mu(x_+, s_+) = \mu^+$. This corrector step completes the iteration; we are back in the desired neighborhood with an appropriate reduction of μ . In summary, we have the following algorithm.

ALGORITHM 6.3 (predictor-corrector method I). *Let $\beta = 1/10$ and $\bar{\beta} = 1/6$, and choose $\zeta > 0$. Given $(x_0, s_0) \in \mathcal{N}(\beta)$, set $k = 0$;*

While $\langle s_k, x_k \rangle > \zeta$ do

begin

set $(x, y, s) := (x_k, y_k, s_k)$ and $\mu := \mu(x, s)$;

compute the affine-scaling direction (p_x, p_y, p_s) ;

choose the largest stepsize α so that (x_+, y_+, s_+) defined by (6.15) lies in $\mathcal{N}(\bar{\beta})$;

set $(x, y, s) := (x_+, y_+, s_+)$;

compute the centering direction (p_x, p_y, p_s) ;

set $(x_{k+1}, y_{k+1}, s_{k+1}) := (x - d_x, y - d_y, s - d_s)$;

$k := k + 1$;

end.

The analysis in this section has only used the local proximity measure λ_2 and the resulting narrow neighborhood $\mathcal{N}(\beta)$. In the next section we will consider a predictor-corrector algorithm based on a possibly much wider neighborhood that uses the proximity measure γ_F . However, for small values of λ_2 and γ_F , there is a close relationship between these measures as shown by Theorem 4.2. To conclude this section, we use this relationship to obtain a lower bound on the step size α^* in the predictor-corrector algorithm above; this will allow us to apply the results of the next section to get more information about α^* .

Suppose that (x, y, s) satisfies (6.2) with $\beta = 1/10$. Then $\lambda_\infty(x, s) \leq \lambda_2(x, s) \leq 1/10$, so that (4.23) implies

$$\gamma_F(x, s) \leq \frac{\left(\frac{1}{10}\right)^2}{\left[1 - \frac{1}{10}\right]} = \frac{1}{90} = .011\dots$$

Let $\tilde{\alpha}$ be the largest α so that the point defined by (6.15) satisfies

$$\gamma_F(x^+, s^+) \leq \frac{1}{6} - \ln\left(1 + \frac{1}{6}\right) = .012\dots$$

Then by (4.19) we see that $\lambda_2(x^+, s^+) \leq 1/6$, and hence

$$\alpha^* \geq \tilde{\alpha}.$$

Thus, in addition to $\alpha^* \geq (10\sqrt{\nu})^{-1}$ from Theorem 6.5, we know that α^* is at least the step size $\tilde{\alpha}$ needed to increase γ_F from $1/90$ to $1/6 - \ln(1 + 1/6)$ along the affine-scaling direction. The next section gives some lower bounds for $\tilde{\alpha}$, which give a fortiori lower bounds on α^* .

We have indicated above that we could have used λ instead of λ_2 to define our neighborhood. The argument of the previous paragraph shows that similar results hold when γ_F is employed. Indeed, Theorem 4.2 suggests that, for algorithms for which every iterate (including intermediate ones) lies in a narrow neighborhood of the central path, similar results and estimates are possible whichever of λ , λ_2 , γ_F , or γ_G is used to define the neighborhood.

7. Functional proximity path-following scheme. In this section we consider an algorithm which generates the main sequence of points in the following neighborhood of the primal-dual central path:

$$\mathcal{F}(\beta) := \{(x, y, s) \in S^0(P) \times S^0(D) : \gamma_F(x, s) \leq \beta\},$$

where $0 < \beta < 1 - \ln 2$. For that purpose it forms first a predicted point in the neighborhood $\mathcal{F}(\beta + \Delta)$, $\Delta > 0$. Then it computes a new point in the initial neighborhood $\mathcal{F}(\beta)$ using the Newton process (5.25). Note that the predictor step size parameter Δ can be arbitrarily large, which contrasts strongly with the predictor-corrector method of the previous section.

Suppose we are given a positive constant β , $0 < \beta < 1 - \ln 2$, and a strictly feasible point $(x_0, y_0, s_0) \in \mathcal{F}(\beta)$. Consider the following predictor-corrector scheme.

ALGORITHM 7.1 (predictor-corrector method II). *Choose $\beta > 0$, $\Delta > 0$, and $\zeta > 0$. Given $(x_0, y_0, s_0) \in \mathcal{F}(\beta)$, set $k = 0$;*

While $\langle s_k, x_k \rangle > \zeta$ do

begin

set $(x, y, s) := (x_k, y_k, s_k)$;

compute the affine-scaling direction (p_x, p_y, p_s) ;

choose the largest stepsize $\alpha := \alpha_k$ so that $(x^+, y^+, s^+) := (x - \alpha p_x, x - \alpha p_x, x - \alpha p_x)$ satisfies $\gamma_F(x^+, s^+) = \beta + \Delta$;

set $(x, y, s) := (x^+, y^+, s^+)$;

compute $(x_{k+1}, y_{k+1}, s_{k+1})$ using the Newton method (5.25) starting from (x, y, s) and terminating as soon as a point in $\mathcal{F}(\beta)$ is found;

$k := k + 1$;

end.

THEOREM 7.1. *Let $\beta = \kappa - \ln(1 + \kappa)$ for some $\kappa \in (0, 1)$. Then,*
 1. *Algorithm 7.1 decreases the duality gap at a linear rate:*

$$(7.1) \quad \langle s_{k+1}, x_{k+1} \rangle \leq (1 - \alpha_k) \langle s_k, x_k \rangle,$$

where

$$(7.2) \quad \frac{\alpha_k^2}{1 - \alpha_k} \geq \frac{1}{\nu} \omega_1(\Delta, \kappa).$$

Moreover,

$$(7.3) \quad \frac{\alpha_k^2}{1 - \alpha_k} \geq \frac{\omega_2(\Delta, \kappa)}{\|p_{x_k} \|_{x_k} \cdot \|p_{s_k} \|_{s_k}} \geq \frac{\omega_2(\Delta, \kappa)}{\sqrt{\nu} \|p_{x_k} \|_{x_k} \cdot \|p_{s_k} \|_{s_k}}.$$

In the above inequalities the positive constants ω_i depend on Δ and κ only.

2. *The number of Newton steps N_k in the corrector phase 1(c) is bounded by an absolute constant:*

$$(7.4) \quad N_k \leq \frac{\Delta}{\bar{\tau} - \ln(1 + \bar{\tau})},$$

where

$$\bar{\tau} = \frac{1}{2} \sqrt{\frac{3\beta}{1 + \beta}}.$$

Proof. In order to simplify the notation let us omit subscripts for the current iterate and use a subscript $+$ for the next iterate; the superscript $+$ denotes the intermediate iterates given by the predictor step.

From Theorem 5.3 we know that

$$(7.5) \quad \Delta = \gamma_F(x^+, s^+) - \gamma_F(x, s) = F(x + \xi p(x, s)) - F(x),$$

where $\xi := \alpha^2/(1 - \alpha)$. Since $\kappa < 1$, in view of Theorem 4.2 we have

$$\lambda_\infty^+(x, s) \leq \lambda_\infty(x, s) \leq \lambda_2(x, s) \leq \kappa,$$

$$\gamma_\infty(x, s) \leq \frac{\kappa}{1 - \kappa}, \quad \gamma_G \leq \frac{\kappa^2}{1 - \kappa}.$$

Therefore, from Lemma 5.4 we get the following inequalities:

$$\langle F'(x), p(x, s) \rangle \leq \frac{2\kappa(1 + \kappa)}{\sqrt{1 - \kappa}} \|p_x \|_x \cdot \|p_s \|_s,$$

$$\langle F'(x), p(x, s) \rangle \leq \frac{\nu \sqrt{\beta}}{1 - \kappa},$$

$$\|p(x, s) \|_x \leq \|p_x \|_x \cdot \|p_s \|_s \leq \frac{\nu}{2(1 - \kappa)}.$$

In view of (7.5), T_* 4.2, and P_* 4.1 (iii),

$$\Delta = F(x + \xi p(x, s)) - F(x) \leq \xi \langle F'(x), p(x, s) \rangle - \xi \|p(x, s) \|_x - \ln(1 - \xi \|p(x, s) \|_x).$$

Hence,

$$\Delta \leq \frac{\xi\nu\sqrt{\beta}}{1-\kappa} - \frac{\xi\nu}{2(1-\kappa)} - \ln\left(1 - \frac{\xi\nu}{2(1-\kappa)}\right).$$

Therefore,

$$\frac{\nu\alpha^2}{1-\alpha} = \xi\nu \geq \omega_1(\Delta, \kappa),$$

where the positive constant ω_1 depends on Δ and κ only. Since from (5.20) and (5.4)

$$\langle s_+, x_+ \rangle = \langle s^+, x^+ \rangle = (1-\alpha)\langle s, x \rangle,$$

we get (7.1) with the estimate (7.2) for the predictor step size.

Similarly,

$$\Delta \leq \frac{2\kappa(1+\kappa)\bar{\xi}}{\sqrt{1-\kappa}} - \bar{\xi} - \ln(1-\bar{\xi}),$$

where

$$\bar{\xi} := \xi \mid p_x \mid_x \cdot \parallel p_s \parallel_s.$$

Therefore,

$$\xi \geq \frac{\omega_2(\Delta, \kappa)}{\parallel p_x \mid_x \cdot \parallel p_s \parallel_s},$$

where the positive constant ω_2 depends on Δ and κ only. Thus,

$$\frac{\alpha^2}{1-\alpha} \geq \frac{\omega_2(\Delta, \kappa)}{\parallel p_x \mid_x \cdot \parallel p_s \parallel_s} \geq \frac{\omega_2(\Delta, \kappa)}{\sqrt{\nu} \parallel p_x \mid_x \cdot \parallel p_s \parallel_s}.$$

Let us now prove that the number of the corrector steps is bounded by an absolute constant.

Let $(\tilde{x}, \tilde{y}, \tilde{s})$ be an intermediate point of the corrector process. In view of Theorem 5.7, the value of the proximity measure is reduced during this step by $\tilde{\tau} - \ln(1 + \tilde{\tau})$, where

$$\tilde{\tau} = \sqrt{\frac{\gamma_G(\tilde{x}, \tilde{s})}{1 + \gamma_\infty(\tilde{x}, \tilde{s})}}.$$

Note also that

$$\tilde{\tau} \geq \bar{\tau} := \sqrt{\frac{3\beta}{4(1+\beta)}},$$

since otherwise by (4.28)

$$\gamma_G(\tilde{x}, \tilde{s}) = \tilde{\tau}^2(1 + \gamma_\infty(\tilde{x}, \tilde{s})) \leq \frac{4}{3}\tilde{\tau}^2(1 + \gamma_G(\tilde{x}, \tilde{s})) \leq \frac{\beta}{1+\beta}(1 + \gamma_G(\tilde{x}, \tilde{s})),$$

and therefore $\gamma_F(\tilde{x}, \tilde{s}) \leq \gamma_G(\tilde{x}, \tilde{s}) \leq \beta$. Thus, we conclude that the number of the corrector steps does not exceed $\Delta/(\bar{\tau} - \ln(1 + \bar{\tau}))$. $\square$

Thus, we have proved three different inequalities for the size of the predictor step. The first inequality (7.2) provides us with the theoretical efficiency estimate of the method. It shows that after

$$O\left(\sqrt{\nu} \ln \frac{\langle s_0, x_0 \rangle}{\epsilon}\right)$$

predictor steps we will generate a feasible primal-dual point with the duality gap less than ϵ . However, this inequality cannot describe the potential “adaptivity” of the scheme. We can get this information from the second inequality (7.3).

To explain this, let us suppose that at some iteration $\alpha_k \leq .9$; if not, then we know that the duality gap decreases by a factor of 10, which is already very good progress. Then (7.3) gives

$$(7.6) \quad 10\alpha_k^2 \geq \frac{\omega_2(\Delta, \kappa)}{\|p_{x_k}\|_{x_k} \cdot \|p_{s_k}\|_{s_k}} \geq \frac{\omega_2(\Delta, \kappa)}{\sqrt{\nu} \|p_{x_k}\|_{x_k} \cdot \|p_{s_k}\|_{s_k}}.$$

Since our algorithm is symmetric, we can also replace the denominator in the second term by $\|p_{x_k}\|_{x_k} \cdot \|p_{s_k}\|_{s_k}$. For definiteness, let us suppose that $\eta_k := \|p_{s_k}\|_{s_k} \geq \|p_{x_k}\|_{x_k}$ (otherwise, x and s can be switched in what follows), and scale the affine-scaling directions by η_k , to get

$$\bar{p}_{x_k} = \frac{1}{\eta_k} p_{x_k}, \quad \bar{p}_{s_k} = \frac{1}{\eta_k} p_{s_k}, \quad \bar{p}_{y_k} = \frac{1}{\eta_k} p_{y_k}.$$

The corresponding step size is

$$\bar{\alpha}_k := \alpha_k \eta_k.$$

Let us call

$$\bar{\alpha}_k^* := \max\{\alpha : x_k \pm \alpha \bar{p}_{x_k} \in K, s_k \pm \alpha \bar{p}_{s_k} \in K^*\} = \frac{1}{\max\{\|\bar{p}_{x_k}\|_{x_k}, \|\bar{p}_{s_k}\|_{s_k}\}} \geq 1$$

the *maximal feasible* step size, where the inequality follows from $\|\bar{p}_{x_k}\|_{x_k} = \|p_{x_k}\|_{x_k} / \eta_k \leq 1$ and similarly $\|\bar{p}_{s_k}\|_{s_k} \leq 1$.

Now the first inequality in (7.6) gives

$$10\bar{\alpha}_k^2 \geq \frac{\omega_2(\Delta, \kappa) \eta_k^2}{\|p_{x_k}\|_{x_k} \cdot \|p_{s_k}\|_{s_k}} = \frac{\omega_2(\Delta, \kappa) \eta_k}{\|p_{x_k}\|_{x_k}} = \frac{\omega_2(\Delta, \kappa)}{\|\bar{p}_{x_k}\|_{x_k}} \geq \omega_2(\Delta, \kappa) \bar{\alpha}_k^*,$$

so $\bar{\alpha}_k = \Omega(\sqrt{\bar{\alpha}_k^*})$; the predictor step size is at least proportional to the square root of the maximal feasible step size in this normalization.

The second inequality in (7.6) gives

$$10\bar{\alpha}_k^2 \geq \frac{\omega_2(\Delta, \kappa) \eta_k^2}{\sqrt{\nu} \|p_{x_k}\|_{x_k} \cdot \|p_{s_k}\|_{s_k}} = \frac{\omega_2(\Delta, \kappa)}{\sqrt{\nu} \|\bar{p}_{x_k}\|_{x_k} \cdot \|\bar{p}_{s_k}\|_{s_k}} \geq \frac{\omega_2(\Delta, \kappa)}{\sqrt{\nu}} (\bar{\alpha}_k^*)^2,$$

so that $\bar{\alpha}_k = \Omega(\nu^{-1/4} \bar{\alpha}_k^*)$; the predictor step size is at least proportional to the maximal feasible step size divided by $\nu^{1/4}$.

These two results provide a strong indication that the functional neighborhood permits large steps.

Note that these bounds also apply to the step size in the λ_2 -based predictor-corrector algorithm of the previous section. Indeed, we showed at the end of section

6 that the step size α^* used there was at least the step size $\tilde{\alpha}$ needed to go from a point in $\mathcal{F}(\beta)$ with $\beta = 1/90 = .011\dots < 1 - \ln 2$ along the affine-scaling direction to a point (x^+, y^+, s^+) with $\gamma_F(x^+, s^+) = \beta^+ := 1/6 - \ln(1 + 1/6) = .012\dots$. Thus the bounds established above also hold for α^* , where now we use $\Delta := \beta^+ - \beta > 0$ and κ such that $\beta = \kappa - \ln(1 + \kappa)$. However, the constants hidden in the Ω s above are much smaller for α^* due to the small value of Δ , while in the predictor-corrector method using the functional proximity neighborhood, Δ can be any positive constant.

REFERENCES

- [ET76] I. EKELAND AND R. TEMAM, *Convex Analysis and Variational Problems*, North-Holland, Amsterdam, 1976.
- [Gu96] O. GÜLER, *Barrier functions in interior-point methods*, Math. Oper. Res., 21 (1996), pp. 860–885.
- [Ka84] N. K. KARMAKAR, *A new polynomial-time algorithm for linear programming*, Combinatorica, 4 (1984), pp. 373–395.
- [KMY89] M. KOJIMA, S. MIZUNO, AND A. YOSHISE, *A polynomial-time algorithm for a class of linear complementarity problems*, Math. Programming, 44 (1989), pp. 1–26.
- [MTY90] S. MIZUNO, M. J. TODD, AND Y. YE, *On adaptive step primal-dual interior-point algorithms for linear programming*, Math. Oper. Res., 18 (1993), pp. 964–981.
- [MYK89] S. MIZUNO, A. YOSHISE, AND T. KIKUCHI, *Practical polynomial time algorithms for linear complementarity problems*, J. Oper. Res. Soc. Japan, 32 (1989), pp. 75–92.
- [MA89] R. D. C. MONTEIRO AND I. ADLER, *Interior path following primal-dual algorithms: Part I: Linear programming*, Math. Programming, 44 (1989), pp. 27–41.
- [Ne96] YU. E. NESTEROV, *Long-step strategies in interior-point primal-dual methods*, Math. Programming, 76 (1996), pp. 47–94.
- [NN94] YU. E. NESTEROV AND A. S. NEMIROVSKII, *Interior Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, PA, 1994.
- [NT97] YU. E. NESTEROV AND M.J. TODD, *Self-scaled barriers and interior-point methods for convex programming*, Math. Oper. Res., 22 (1997), pp. 1–42.
- [Ro60] O. ROTHBAUS, *Domains of positivity*, Abh. Math. Sem. Univ. Hamburg, 24 (1960), pp. 189–235.
- [TTT96] M.J. TODD, K.-C. TOH, AND R. H. TÜTÜNCÜ, *On the Nesterov-Todd Direction in Semidefinite Programming*, SIAM J. Optim., 8 (1998), to appear.
- [Tu96] L. TUNÇEL, *Primal-Dual Symmetry and Scale Invariance of Interior-Point Algorithms for Convex Optimization*, Research report CORR 96–18, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, ON, Canada, 1996.

ON EXTENDING SOME PRIMAL–DUAL INTERIOR-POINT ALGORITHMS FROM LINEAR PROGRAMMING TO SEMIDEFINITE PROGRAMMING*

YIN ZHANG†

Abstract. This work concerns primal–dual interior-point methods for semidefinite programming (SDP) that use a search direction originally proposed by Helmberg et al. [*SIAM J. Optim.*, 6 (1996), pp. 342–361] and Kojima, Shindoh, and Hara [*SIAM J. Optim.*, 7 (1997), pp. 86–125.] and recently rediscovered by Monteiro [*SIAM J. Optim.*, 7 (1997), pp. 663–678] in a more explicit form. In analyzing these methods, a number of basic equalities and inequalities were developed in [Kojima, Shindoh, and Hara] and also in [Monteiro] through different means and in different forms.

In this paper, we give a concise derivation of the key equalities and inequalities for complexity analysis along the exact line used in linear programming (LP), producing basic relationships that have compact forms almost identical to their counterparts in LP. We also introduce a new formulation of the central path and variable-metric measures of centrality. These results provide convenient tools for deriving polynomiality results for primal–dual algorithms extended from LP to SDP using the aforementioned and related search directions. We present examples of such extensions, including the long-step infeasible-interior-point algorithm of Zhang [*SIAM J. Optim.*, 4 (1994), pp. 208–227].

Key words. semidefinite programming, primal–dual interior-point methods

AMS subject classifications. 15A15, 15A09, 15A23

PII. S1052623495296115

1. Introduction. In this paper, we consider primal–dual interior-point algorithms for solving the semidefinite program (SDP)

$$(1.1) \quad \begin{array}{ll} \min & C \bullet X \\ \text{subject to (s.t.)} & A_i \bullet X = b_i, \quad i = 1, 2, \dots, m, \\ & X \geq 0, \end{array}$$

where C , X , and A_i are symmetric matrices in $\mathbf{R}^{n \times n}$, $b_i \in \mathbf{R}$, $C \bullet X = \text{tr}(CX)$, and $X \geq 0$ means that X is positive semidefinite.

For any $m \times n$ matrix A , $\text{vec } A$ denotes the mn -vector obtained from stacking the columns of A one by one from the first to the last. The norm $\|\cdot\|$ is by default the 2-norm for vectors and spectral norm for matrices unless specified otherwise, such as $\|\cdot\|_F$ for Frobenius norm. In this paper, we will use extensively Kronecker-product notations (see Appendix A).

Under certain assumptions, the optimality conditions for (1.1) are

$$(1.2) \quad \begin{pmatrix} \mathbf{A}^T y + \text{vec } Z - \text{vec } C \\ \mathbf{A}(\text{vec } X) - b \\ XZ \end{pmatrix} = 0, \quad X, Z \geq 0,$$

where $\mathbf{A}^T = [\text{vec } A_1 \text{ vec } A_2 \cdots \text{vec } A_m]$, $b^T = [b_1 \ b_2 \ \cdots \ b_m]$, $y \in \mathbf{R}^m$, and $Z \in \mathbf{R}^{n \times n}$ are the dual variables. The zero on the right-hand side of the equation in (1.2) means that every term on the left-hand side is zero.

*Received by the editors December 8, 1995; accepted for publication (in revised form) January 27, 1997. This research was supported in part by US Department of Energy grant DE-FG02-93ER25171-A001.

<http://www.siam.org/journals/siopt/8-2/29611.html>

†Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005 (zhang@caam.rice.edu).

We say that a point (X, y, Z) is feasible if it satisfies the first two linear equations in (1.2), $X \geq 0$ and $Z \geq 0$, and strictly feasible if in addition X and Z are symmetric positive definite (s.p.d.). It is known that if a strictly feasible point exists, then (1.2) has a solution. If we require that all the matrices in SDP be diagonal, then (1.1) reduces to the familiar standard form linear program.

SDP arises in many scientific and engineering fields. The recent book by Boyd et al. [4] contains many examples from system and control theory. Many applications in combinatorial optimization are included in [1]. In addition, many problems in SDP come from eigenvalue optimization (see [18, 19], for example).

SDP has recently attracted active research from the interior-point methods community. A rather comprehensive list of references for research works and applications in this field up to early 90s can be found in [1]. Some more recent works include [3, 5, 7, 12, 16, 22, 23].

Primal–dual interior-point methods, especially the infeasible ones, have proven to be the most efficient methods in LP¹, and many polynomiality results exist for these methods. Extending existing results to SDP is a promising area from both theoretical and practical points of view. Recent work on primal–dual methods for SDP include [2, 11, 17, 24].

The analysis for SDP has appeared to be more difficult than that for LP. A large part of the theoretical difficulty has to do with the issue of maintaining symmetry in linearized complementarity, as will be discussed in the next section. Very recently, Monteiro [15] proposed two closely related linearized complementarity equations and used the first one to successfully extend two feasible-interior-point primal–dual algorithms to SDP, including the Kojima–Mizuno–Yoshise [10] long-step feasible algorithm. It was pointed out by Kojima [8] that the search directions of Monteiro are in fact special cases of those proposed earlier by Kojima, Shindoh, and Hara [11], and one of the directions was also proposed independently by Helmberg et al. [5]. In [11], the authors analyzed a short-step path-following algorithm and an infeasible-interior-point short-step potential-reduction algorithm. Consequently, some of the results in [15] coincide with those in [11] but with alternative and shorter proofs. Nevertheless, Monteiro’s more explicit formulation does offer certain advantage over the formulation of Kojima, Shindoh, and Hara. As we will demonstrate in this paper, Monteiro’s formulation reveals some deep similarity between LP and SDP and allows insightful complexity analysis for primal–dual methods for SDP.

Based on Monteiro’s second linearized complementarity equation, which we prefer to the first one for computational reasons, we first derive in this paper a set of key equalities and inequalities central to the analysis of primal–dual methods for SDP. These equalities and inequalities are not completely new in the sense that they ultimately have similar functionalities as those derived by Kojima, Shindoh, and Hara [11] (and some of those by Monteiro [15] due to the aforementioned equivalence). However, our concise analysis follows the exact line of analysis for LP and produces familiar and compact results.

In this paper, we also introduce a new formulation of the central path and a variable-metric measure of centrality. Together these results make the task of extending polynomiality results to SDP a much amenable process for many existing algorithms. We present some examples of such easy extensions.

¹In this paper we will use LP as the representative of the class of problems with the common basic characteristics in terms of primal–dual methods, including convex quadratic programming, monotone linear complementarity problem, and so on.

Our analysis starts from the following simple identity which opens the door to the entire development of the paper.

PROPOSITION 1.1. *Let vectors $u, v, r \in \mathbf{R}^p$ and nonsingular matrices $E, F \in \mathbf{R}^{p \times p}$ satisfy*

$$Eu + Fv = r.$$

If $S = FE^T$ is s.p.d., then

$$(1.3) \quad \|D^{-T}u\|^2 + \|Dv\|^2 + 2u \cdot v = \|S^{-\frac{1}{2}}r\|^2,$$

where

$$(1.4) \quad D = S^{-\frac{1}{2}}F = S^{\frac{1}{2}}E^{-T} \quad \text{and} \quad D^{-T} = S^{-\frac{1}{2}}E = S^{\frac{1}{2}}F^{-T}.$$

Proof. Premultiply both sides of $Eu + Fv = r$ by $S^{-\frac{1}{2}}$, and then take 2-norm squared. $\square$

This paper is organized as follows. In section 2, we discuss the issue of linearizing the complementarity condition. Section 3 contains our derivations of basic equalities and inequalities. In section 4, we introduce a new definition of central path and two of its neighborhoods suitable for SDP and present some useful properties. Section 5 is a brief discussion on the role played by the condition number of $Z^{\frac{1}{2}}XZ^{\frac{1}{2}}$ in determining the order of iteration-complexity. Section 6 contains some complexity results as examples to demonstrate the power of the basic equalities and inequalities in extending polynomiality results to SDP. The paper concludes with some remarks in section 7.

2. Linearization. Since primal-dual methods are Newton-like methods applied to the optimality system (1.2), at a given point (X, y, Z) , an increment direction $(\Delta X, \Delta y, \Delta Z)$ should satisfy

$$(2.1) \quad \mathbf{A}^T \Delta y + \mathbf{vec} \Delta Z = \mathbf{vec} R_d \equiv -(\mathbf{A}^T y + \mathbf{vec} Z - \mathbf{vec} C)$$

and

$$(2.2) \quad \mathbf{Avec} \Delta X = r_p \equiv -(\mathbf{Avec} X - b).$$

Equivalently, one can write (2.1) as

$$(2.3) \quad \sum_{i=1}^m \Delta y_i A_i + \Delta Z = R_d = -\left(\sum_{i=1}^m y_i A_i + Z - C \right).$$

Hence ΔZ will be symmetric whenever Z is.

The linearization of the complementarity condition $X_+ Z_+ = 0$ at (X, Z) , or more generally the perturbed complementarity condition $X_+ Z_+ = \sigma \mu I$ for $\sigma \in [0, 1]$, where

$$\mu = X \bullet Z / n$$

is the so-called duality gap², has proven to be a delicate issue due to the need of maintaining symmetry in ΔX . The most obvious linearization is

$$(2.4) \quad XZ + \Delta X Z + X \Delta Z = \sigma \mu I.$$

²Strictly speaking, μ is the duality gap only when (X, y, Z) is feasible for some y . But for convenience we will use the term regardless of feasibility. Moreover, since not all semidefinite programs have equal primal and dual optimal objective values, the term “duality gap” is also used, in that context, to refer to the difference between the two.

The new duality gap at the updated point

$$(X_+, y_+, Z_+) = (X, y, Z) + \alpha(\Delta X, \Delta y, \Delta Z)$$

will be

$$\begin{aligned}\mu_+ &= (X + \alpha\Delta X) \bullet (Z + \alpha\Delta Z)/n \\ &= (1 - \alpha)\mu + \alpha(X \bullet Z + Z \bullet \Delta X + X \bullet \Delta Z)/n \\ &\quad + \alpha^2(\Delta X \bullet \Delta Z)/n.\end{aligned}$$

If $(\Delta X, \Delta Z)$ satisfies (2.4), we have

$$(2.5) \quad X \bullet Z + Z \bullet \Delta X + X \bullet \Delta Z = n\sigma\mu.$$

Therefore,

$$(2.6) \quad \mu_+ = (1 - \alpha + \sigma\alpha)\mu + \alpha^2(\Delta X \bullet \Delta Z)/n.$$

It is critical to realize that (2.5) and (2.6) still hold true if the equation (2.4) is *similarly* transformed and symmetrized. Let us define the similarly transformed symmetrization operator

$$(2.7) \quad H_P(M) = \frac{1}{2} [PMP^{-1} + (PMP^{-1})^T]$$

for a given nonsingular matrix P . Therefore, (2.5) and (2.6) hold true if $(\Delta X, \Delta Z)$ satisfies the following similarly transformed and symmetrized equation:

$$(2.8) \quad H_P(XZ + \Delta XZ + X\Delta Z) = \sigma\mu I.$$

For interior-point methods, it is imperative to maintain symmetry of the X matrices, which in turn requires symmetry of ΔX at each iteration. Since the three matrix products in the left-hand side of (2.4) are generally nonsymmetric, neither will be ΔX in (2.4). On the other hand, ΔX in (2.8) does enjoy symmetry thanks to the symmetrization.

In [2], Alizadeh, Haerberly, and Overton used pure symmetrization without a similar transformation, i.e., they used $P = I$ in equation (2.8), leading to

$$(2.9) \quad (Z\Delta X + \Delta XZ) + (X\Delta Z + \Delta ZX) = 2\sigma\mu I - XZ - ZX.$$

Using the symmetrized linearized-complementarity equation (2.9), Alizadeh, Haerberly, and Overton presented promising numerical results in [2] showing that the algorithm based on (2.9) is the most robust among a number of primal-dual algorithms. However, (2.9) does seem to have its difficulties. Since Proposition 1.1 does not hold for the matrices in (2.9), our analysis in this paper does not seem to be applicable to search directions resulting from (2.9). Moreover, (2.9) necessitates solving Lyapunov equations, i.e., equations of the form $AX + XA = B$ for matrix X (see [2] for more details).

Recently, Monteiro [15] proposed to use $P = X^{-\frac{1}{2}}$ or $P = Z^{\frac{1}{2}}$ in (2.8) for the similar transformations. He analyzed some primal-dual feasible-interior-point algorithms using the first choice $P = X^{-\frac{1}{2}}$ and established polynomial complexity for them. Monteiro's linearized complementarity equations, special cases of those of Kojima, Shindoh, and Hara in more explicit forms, motivated our analysis in this paper.

Out of the two scaling matrices proposed by Monteiro, we prefer the second one, $P = Z^{\frac{1}{2}}$, to the first one for computational reasons that will be made clear at the end of this section. From now on, we omit the subscript P from H_P in (2.7) whenever $P = Z^{\frac{1}{2}}$.

Substituting $P = Z^{\frac{1}{2}}$ into (2.8), and premultiplying and postmultiplying the resulting equation $H(XZ + X\Delta Z + \Delta XZ) = \sigma\mu I$ by $\sqrt{2}Z^{\frac{1}{2}}$, we obtain

$$(2.10) \quad 2Z(\Delta X)Z + ZX(\Delta Z) + (\Delta Z)XZ = R_c,$$

where

$$(2.11) \quad R_c = 2(\sigma\mu Z - ZXZ) = 2Z^{\frac{1}{2}}(\sigma\mu I - Z^{\frac{1}{2}}XZ^{\frac{1}{2}})Z^{\frac{1}{2}}.$$

Equation (2.10) can also be obtained from linearizing, at (X, Z) , the equation

$$\frac{1}{2} [(ZX_+Z_+) + (ZX_+Z_+)^T] = \sigma\mu Z,$$

i.e., the symmetrization of the equation $Z(X_+Z_+ - \sigma\mu I) = 0$. Since Z varies from iteration to iteration, the direction $(\Delta X, \Delta y, \Delta Z)$ resulting from using (2.10) is not equivalent to a Newton direction for a fixed nonlinear system.

In Kronecker-product notation, (2.10) is

$$(2.12) \quad 2(Z \otimes Z)\text{vec}\Delta X + (ZX \otimes I + I \otimes ZX)\text{vec}\Delta Z = \text{vec}R_c.$$

Let us define

$$(2.13) \quad E = 2Z \otimes Z$$

and

$$(2.14) \quad F = ZX \otimes I + I \otimes ZX.$$

Then (2.12) becomes

$$(2.15) \quad E\text{vec}\Delta X + F\text{vec}\Delta Z = \text{vec}R_c.$$

It is worth noting that E is s.p.d., but F is generally nonsymmetric.

With the linearized complementarity (2.15), together with (2.1) and (2.2), we have the linear system satisfied by the increment $(\Delta X, \Delta y, \Delta Z)$ at point (X, y, Z) :

$$(2.16) \quad \begin{bmatrix} 0 & \mathbf{A}^T & I \\ \mathbf{A} & 0 & 0 \\ E & 0 & F \end{bmatrix} \begin{pmatrix} \text{vec}\Delta X \\ \Delta y \\ \text{vec}\Delta Z \end{pmatrix} = \begin{pmatrix} \text{vec}R_d \\ r_p \\ \text{vec}R_c \end{pmatrix}.$$

Just like in LP, one can solve the linear system (2.16) by the following procedure.

Procedure-A:

- (1) $[\mathbf{A}(E^{-1}F)\mathbf{A}^T]\Delta y = r_p + \mathbf{A}[E^{-1}(F\text{vec}R_d - \text{vec}R_c)]$.
- (2) $\Delta Z = R_d - \sum_{i=1}^m \Delta y_i A_i$.
- (3) $\Delta X = \sigma\mu Z^{-1} - X - [X(\Delta Z)Z^{-1} + (X(\Delta Z)Z^{-1})^T]/2$.

End

This procedure was essentially used in both [5] and [11].

It is important to observe that $E^{-1} = (Z^{-1} \otimes Z^{-1})/2$ is relatively easy to compute. This is the reason why out of the two scaling schemes we prefer to use the second scaling, $P = Z^{\frac{1}{2}}$, in (2.7) over the first one, $P = X^{-\frac{1}{2}}$, which would lead to $E = XZ \otimes I + I \otimes XZ$ and necessitate the solution of Lyapunov equations.

In addition to the easy inversion of E , another computational advantage of (2.10) over (2.9) is that Cholesky, rather than LU, factorization can be utilized in the first step of the above procedure, as is indicated by the following simple fact, which was also observed in [5].

PROPOSITION 2.1. *The matrix*

$$E^{-1}F = \frac{(X \otimes Z^{-1} + Z^{-1} \otimes X)}{2}$$

is s.p.d. Therefore $\mathbf{A}(E^{-1}F)\mathbf{A}^T$ is s.p.d. whenever $\mathbf{A}$ has full rank.

Proof. See Appendix A. $\square$

The following proposition states that if the primal equality constraints are consistent, then so is the linear system (2.16). It is worth noting that for the purpose of establishing complexity one does not need to assume that $\mathbf{A}$ is full rank, which we do not in this paper.

PROPOSITION 2.2. *Let X and Z be s.p.d. If b is in the range of $\mathbf{A}$, then equation (1) in Procedure-A is consistent and, consequently, so is (2.16).*

Proof. Observe that under the assumption of the proposition r_p and hence the whole right-hand side of equation (1) are in the range of $\mathbf{A}$. Since the matrix $E^{-1}F$ is s.p.d, we can define $\hat{\mathbf{A}} = \mathbf{A}(E^{-1}F)^{1/2}$ and rewrite equation (1) as

$$\hat{\mathbf{A}}\hat{\mathbf{A}}^T \Delta y = \hat{\mathbf{A}}r$$

for some vector r , which is clearly consistent. Consequently, there exists $(\Delta X, \Delta y, \Delta Z)$ that satisfies (2.16). $\square$

The following proposition can be easily verified for matrices E and F defined in (2.13) and (2.14), respectively. It allows us to use Proposition 1.1, thus paving the way to the development of the paper.

PROPOSITION 2.3. *The matrix $S = FE^T$ is s.p.d. and can be written as*

$$(2.17) \quad S = E^{\frac{1}{2}} \hat{F} E^{\frac{1}{2}},$$

where $\hat{F}$ is s.p.d. given by

$$(2.18) \quad \hat{F} = E^{-\frac{1}{2}} F E^{\frac{1}{2}} = Z^{\frac{1}{2}} X Z^{\frac{1}{2}} \otimes I + I \otimes Z^{\frac{1}{2}} X Z^{\frac{1}{2}}.$$

3. Basic relationships. In this section, we will assume that X and Z are s.p.d. and $(\Delta X, \Delta y, \Delta Z)$ satisfies (2.16). When (X, y, Z) is feasible, i.e., $R_d = 0$ and $r_p = 0$, it is easy to check that $\Delta X \bullet \Delta Z = 0$.

Due to similarity, matrices XZ , ZX , $Z^{\frac{1}{2}} X Z^{\frac{1}{2}}$, and $X^{\frac{1}{2}} Z X^{\frac{1}{2}}$ all have the same spectrum, which we denote by and arrange as

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n > 0.$$

Moreover, we define the spectral decomposition of $Z^{\frac{1}{2}} X Z^{\frac{1}{2}}$ as

$$(3.1) \quad Z^{\frac{1}{2}} X Z^{\frac{1}{2}} = Q \Lambda Q^T,$$

where $Q^T Q = I$ and

$$\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n).$$

Before we proceed, let us recall that

$$\mu = X \bullet Z/n = \sum_{i=1}^n \lambda_i/n$$

and S , D , and R_c are defined by (2.17), (1.4), and (2.11), respectively.

3.1. General relationships.

LEMMA 3.1.

$$\|D^{-T} \text{vec} \Delta X\|^2 + \|D \text{vec} \Delta Z\|^2 + 2\Delta X \bullet \Delta Z = \|S^{-\frac{1}{2}} \text{vec} R_c\|^2.$$

Proof. The equality follows immediately from Propositions 1.1 and 2.3. $\square$

LEMMA 3.2.

$$\|S^{-\frac{1}{2}} \text{vec} R_c\|^2 = \sum_{i=1}^n \frac{(\sigma\mu - \lambda_i)^2}{\lambda_i} = \left(1 - 2\sigma + \sigma^2 \frac{\mu}{n} \sum_{i=1}^n \frac{1}{\lambda_i}\right) n\mu.$$

Moreover, if $\lambda_n \geq \gamma\mu$ for $\gamma \in (0, 1)$, then

$$\|S^{-\frac{1}{2}} \text{vec} R_c\|^2 \leq \min \left(\frac{\|\Lambda - \sigma\mu I\|_F^2}{\gamma\mu}, (1 - 2\sigma + \sigma^2/\gamma)n\mu \right).$$

Proof. Observe that

$$(3.2) \quad \text{vec} R_c = \sqrt{2}E^{\frac{1}{2}}(Q \otimes Q) \text{vec}(\sigma\mu I - \Lambda)$$

and

$$(3.3) \quad \hat{F} = (Q \otimes Q)(\Lambda \otimes I + I \otimes \Lambda)(Q^T \otimes Q^T).$$

The proof of the first part follows directly from the fact that

$$\|S^{-\frac{1}{2}} \text{vec} R_c\|^2 = (\text{vec} R_c)^T S^{-1} (\text{vec} R_c),$$

Proposition 2.3, (3.2), and (3.3). The second part is obvious. $\square$

LEMMA 3.3.

$$\|H(\Delta X \Delta Z)\|_F \leq \|Z^{\frac{1}{2}}(\Delta X \Delta Z)Z^{-\frac{1}{2}}\|_F \leq \frac{\sqrt{\kappa}}{2} (\|D^{-T} \text{vec} \Delta X\|^2 + \|D \text{vec} \Delta Z\|^2),$$

where $\kappa = \lambda_1/\lambda_n$ is the spectral condition number of the matrix $Z^{\frac{1}{2}} X Z^{\frac{1}{2}}$.

Proof. We start with

$$(3.4) \quad \|H(\Delta X \Delta Z)\|_F \leq \|Z^{\frac{1}{2}}(\Delta X \Delta Z)Z^{-\frac{1}{2}}\|_F \leq \|Z^{\frac{1}{2}}(\Delta X)Z^{\frac{1}{2}}\|_F \|Z^{-\frac{1}{2}}(\Delta Z)Z^{-\frac{1}{2}}\|_F.$$

Consider

$$\begin{aligned} \|D^{-T} \text{vec} \Delta X\|^2 &= (\text{vec} \Delta X)^T E S^{-1} E (\text{vec} \Delta X) \\ &= (\text{vec} \Delta X)^T E^{\frac{1}{2}} \hat{F}^{-1} E^{\frac{1}{2}} (\text{vec} \Delta X) \\ &= 2[\text{vec}(Z^{\frac{1}{2}}(\Delta X)Z^{\frac{1}{2}})]^T \hat{F}^{-1} [\text{vec}(Z^{\frac{1}{2}}(\Delta X)Z^{\frac{1}{2}})]. \end{aligned}$$

Since $\hat{F}$ has eigenvalues $\{\lambda_i + \lambda_j\}$ (see (2.18) and Appendix A),

$$\|D^{-T} \mathbf{vec} \Delta X\|^2 \geq \frac{1}{\lambda_1} \|\mathbf{vec}(Z^{\frac{1}{2}}(\Delta X)Z^{\frac{1}{2}})\|^2 = \frac{1}{\lambda_1} \|Z^{\frac{1}{2}}(\Delta X)Z^{\frac{1}{2}}\|_F^2$$

or

$$(3.5) \quad \|Z^{\frac{1}{2}}(\Delta X)Z^{\frac{1}{2}}\|_F \leq \sqrt{\lambda_1} \|D^{-T} \mathbf{vec} \Delta X\|.$$

By similar calculation we have

$$\|D \mathbf{vec} \Delta Z\|^2 \geq \lambda_n \|\mathbf{vec}(Z^{-\frac{1}{2}}(\Delta Z)Z^{-\frac{1}{2}})\|^2$$

or

$$(3.6) \quad \|Z^{-\frac{1}{2}}(\Delta Z)Z^{-\frac{1}{2}}\|_F \leq \frac{1}{\sqrt{\lambda_n}} \|D \mathbf{vec} \Delta Z\|.$$

Combining (3.4), (3.5), and (3.6), we have

$$\|Z^{\frac{1}{2}}(\Delta X \Delta Z)Z^{-\frac{1}{2}}\|_F \leq \sqrt{\frac{\lambda_1}{\lambda_n}} \|D^{-T} \mathbf{vec} \Delta X\| \|D \mathbf{vec} \Delta Z\|.$$

Finally, observe that

$$\|D^{-T} \mathbf{vec} \Delta X\| \|D \mathbf{vec} \Delta Z\| \leq \frac{1}{2} (\|D^{-T} \mathbf{vec} \Delta X\|^2 + \|D \mathbf{vec} \Delta Z\|^2),$$

which completes the proof. $\square$

3.2. Feasible case. First consider that (X, y, Z) is feasible. In this case, $\Delta X \bullet \Delta Z = 0$. Thus the following corollary follows directly from Lemmas 3.1, 3.2, and 3.3.

COROLLARY 3.1. *Let (X, Z) be feasible and $\lambda_n \geq \gamma\mu$ for $\gamma \in (0, 1)$. Then*

$$\|H(\Delta X \Delta Z)\|_F \leq \frac{\sqrt{\kappa}}{2} \min \left(\frac{\|\Lambda - \sigma\mu I\|_F^2}{\gamma\mu}, (1 - 2\sigma + \sigma^2/\gamma)n\mu \right).$$

3.3. Infeasible case. Complexity analysis for infeasible-interior-point algorithms started in [25] where a couple of basic relationships were observed for monotone horizontal linear complementarity problems. These results were later modified into different forms by various authors. For the sake of clarity, we restate some of the original results in the context of LP.

LEMMA 3.4. *Let (x, y, z) , (x_0, y_0, z_0) be such that $(x, z) > 0$, $(x_0, z_0) > 0$ and for $\nu \in [0, 1]$*

$$\begin{aligned} A^T y + z - c &= \nu(A^T y_0 + z_0 - c), \\ Ax - b &= \nu(Ax_0 - b). \end{aligned}$$

Moreover, let $(x, y, z) + (\Delta x, \Delta y, \Delta z)$ and (x_f, y_f, z_f) both satisfy the equations $Ax = b$ and $A^T y + z = c$. Then

$$(3.7) \quad [\Delta x + \nu(x_0 - x_f)] \cdot [\Delta z + \nu(z_0 - z_f)] = 0,$$

$$(3.8) \quad [(x - x_f) - \nu(x_0 - x_f)] \cdot [(z - z_f) - \nu(z_0 - z_f)] = 0;$$

or equivalently,

$$(3.9) \quad \Delta x \cdot \Delta z = -\nu[\Delta x \cdot (z_0 - z_f) + (x_0 - x_f) \cdot \Delta z] - \nu^2(x_0 - x_f) \cdot (z_0 - z_f),$$

$$(3.10)$$

$$\begin{aligned} \nu[x \cdot (z_0 - z_f) + (x_0 - x_f) \cdot z] &= x \cdot z + \nu^2 x_0 \cdot z_0 + \nu(1 - \nu)(x_0 \cdot z_f + x_f \cdot z_0) \\ &\quad - x \cdot z_f - x_f \cdot z + (1 - \nu)^2 x_f \cdot z_f. \end{aligned}$$

The proof is mostly direct substitutions, so we omit it here.

We note that the equalities (3.7) and (3.8) are taken directly from the proofs of Lemmas 6.1 and 6.2 in [25] where they appeared as inequalities in the more general context of monotone linear complementarity problems³.

Furthermore, we assume that

$$(x_f, y_f, z_f) = (x_*, y_*, z_*), \quad x_0 - x_* \geq 0, \quad z_0 - z_* \geq 0,$$

where (x_*, y_*, z_*) is a solution of the linear program, as was done in [25]. Then it follows immediately from (3.10) that, noting $\nu \in [0, 1]$,

$$(3.11) \quad \nu[x \cdot (z_0 - z_*) + (x_0 - x_*) \cdot z] \leq x \cdot z + \nu x_0 \cdot z_0 + \nu(x_0 \cdot z_* + x_* \cdot z_0).$$

To translate the above results to SDP, let us likewise introduce two points: the first point (X_0, y_0, Z_0) , where X_0 and Z_0 are s.p.d., satisfies for some $\nu \in [0, 1]$, and along with (X, y, Z) , the equations

$$(3.12) \quad \mathbf{A}^T y + \mathbf{vec} Z - \mathbf{vec} C = \nu(\mathbf{A}^T y_0 + \mathbf{vec} Z_0 - \mathbf{vec} C),$$

$$(3.13) \quad \mathbf{A}(\mathbf{vec} X) - b = \nu(\mathbf{A}(\mathbf{vec} X_0) - b),$$

and the second point (X_*, y_*, Z_*) is a solution to (1.2).

Under these analogous conditions, the above basic relationships in LP can be trivially extended to SDP after changing the vectors into corresponding matrices along with the change in inner product. Hence to save space we will recycle Lemma 3.4 and inequality (3.11) in our analysis for SDP without copying them into matrix forms.

The following lemma is a direct extension of Lemma 6.2 in [25].

LEMMA 3.5. *For any (X_0, y_0, Z_0) satisfying (3.12) and (3.13), and a solution (X_*, y_*, Z_*) to (1.2),*

$$(3.14) \quad \|D^{-T} \mathbf{vec} \Delta X\|^2 + \|D \mathbf{vec} \Delta Z\|^2 \leq \left(\xi + \sqrt{\xi^2 + \eta} \right)^2,$$

where

$$(3.15) \quad \xi = \nu(\|D^{-T} \mathbf{vec}(X_0 - X_*)\| + \|D \mathbf{vec}(Z_0 - Z_*)\|),$$

$$(3.16) \quad \eta = \|S^{-\frac{1}{2}} R_c\|_F^2 + 2\nu^2(X_0 - X_*) \bullet (Z_0 - Z_*).$$

Proof. First define

$$t = \left(\|D^{-T} \mathbf{vec} \Delta X\|^2 + \|D \mathbf{vec} \Delta Z\|^2 \right)^{\frac{1}{2}}.$$

³Equality (3.7) corresponds to the first inequality after (6.8) in [25] and (3.8) to the one before (6.1) in [25]. The equivalence can be seen after the substitution $u^k = x^k - \nu(x^0 - u^0)$ given in Lemma 4.1 of [25].

From (3.9), we have

$$\begin{aligned}
\Delta X \bullet \Delta Z &= -\nu(D^{-T} \text{vec} \Delta X)^T D \text{vec}(Z_0 - Z_*) \\
&\quad -\nu(D \text{vec} \Delta Z)^T D^{-T} \text{vec}(X_0 - X_*) \\
&\quad -\nu^2(X_0 - X_*) \bullet (Z_0 - Z_*) \\
&\geq -\nu \|D^{-T} \text{vec} \Delta X\| \|D \text{vec}(Z_0 - Z_*)\| \\
&\quad -\nu \|D \text{vec} \Delta Z\| \|D^{-T} \text{vec}(X_0 - X_*)\| \\
&\quad -\nu^2(X_0 - X_*) \bullet (Z_0 - Z_*) \\
&\geq -\nu(\|D \text{vec}(Z_0 - Z_*)\| + \|D^{-T} \text{vec}(X_0 - X_*)\|)t \\
&\quad -\nu^2(X_0 - X_*) \bullet (Z_0 - Z_*) \\
&= -\xi t - \nu^2(X_0 - X_*) \bullet (Z_0 - Z_*).
\end{aligned}$$

Substituting the above and t^2 into the first equation in Lemma 3.1, we obtain

$$t^2 - 2\xi t - \eta \leq 0.$$

The quadratic $\tau^2 - 2\xi\tau - \eta$ has a unique positive root at $\tau_+ = \xi + \sqrt{\xi^2 + \eta}$ and is positive for $\tau > \tau_+$; hence we must have $t \leq \tau_+$, which proves the lemma. $\square$

Now we need to estimate the quantities ξ and η in Lemma 3.5. Let us denote

$$\mu_0 = X_0 \bullet Z_0/n.$$

In the rest of the paper, we will selectively use the following conditions (in addition to (3.12) and (3.13) which we always assume) that are also direct extensions of corresponding conditions in [25]:

$$(3.17) \quad \nu \leq \mu/\mu_0 \leq 1,$$

$$(3.18) \quad X_0 - X_* \geq 0 \quad \text{and} \quad Z_0 - Z_* \geq 0,$$

$$(3.19) \quad X_0 = Z_0 = \rho I,$$

where (X_*, y_*, Z_*) is a solution to the semidefinite program (1.2) and

$$\rho \geq \frac{\text{tr} X_* + \text{tr} Z_*}{n}.$$

We observe that a condition similar to condition (3.17) was first used by Kojima, Megiddo, and Mizuno [9]. We also mention that we impose conditions (3.18) and (3.19) mainly for the sake of simplicity. They can be somewhat weakened by extra work.

Conditions (3.18) and (3.19) are imposed in order to obtain polynomial complexity bounds. In general, a priori information on the size of a solution is not available, thus a very large value for ρ may have to be taken to accommodate the worst-case scenario. This is particularly problematic in SDP where the size of a solution may become excessively large (see [21, Section 1.6, Example 4]). Fortunately, in practice a very large value for ρ does not appear to be necessary for good convergence behavior.

LEMMA 3.6. *Under condition (3.17),*

$$\nu^2(X_0 - X_*) \bullet (Z_0 - Z_*) \leq n\mu.$$

Moreover, if $\lambda_n \geq \gamma\mu$ for some $\gamma \in (0, 1)$, then

$$(3.20) \quad \eta \leq (3 - 2\sigma + \sigma^2/\gamma)n\mu.$$

Proof. The first inequality follows from the calculation

$$\begin{aligned}\nu^2(X_0 - X_*) \bullet (Z_0 - Z_*) &= \nu^2(X_0 \bullet Z_0 - X_0 \bullet Z_* - X_* \bullet Z_0 + X_* \bullet Z_*) \\ &\leq \nu(\mu/\mu_0)(X_0 \bullet Z_0 - X_0 \bullet Z_* - X_* \bullet Z_0) \\ &\leq (\mu/\mu_0)X_0 \bullet Z_0 = n\mu,\end{aligned}$$

where we use the fact that matrices X_* , Z_* , X_0 , and Z_0 are all s.p.d. Inequality (3.20) follows directly from Lemma 3.2 and (3.16). $\square$

LEMMA 3.7. *Under conditions (3.17) and (3.18),*

$$\xi \leq \frac{n\mu}{\sqrt{\lambda_n}} \left(2 + \frac{X_0 \bullet Z_* + Z_0 \bullet X_*}{X_0 \bullet Z_0} \right).$$

In addition, if condition (3.19) also holds and $\lambda_n \geq \gamma\mu$, then

$$\xi \leq 3n\sqrt{\frac{\mu}{\gamma}}.$$

Proof. First we calculate

$$\begin{aligned}\|D^{-T}\mathbf{vec}(X_0 - X_*)\|^2 &= (\mathbf{vec}(X_0 - X_*))^T (ES^{-1}E)\mathbf{vec}(X_0 - X_*) \\ &= (\mathbf{vec}(X_0 - X_*))^T (E^{\frac{1}{2}}\hat{F}^{-1}E^{\frac{1}{2}})\mathbf{vec}(X_0 - X_*) \\ &= 2(\mathbf{vec}Z^{\frac{1}{2}}(X_0 - X_*)Z^{\frac{1}{2}})^T (\hat{F}^{-1})\mathbf{vec}(Z^{\frac{1}{2}}(X_0 - X_*)Z^{\frac{1}{2}}) \\ &\leq (1/\lambda_n)\|\mathbf{vec}(Z^{\frac{1}{2}}(X_0 - X_*)Z^{\frac{1}{2}})\|^2 \\ &= (1/\lambda_n)\|Z^{\frac{1}{2}}(X_0 - X_*)Z^{\frac{1}{2}}\|_F^2 \\ &\leq (1/\lambda_n)[\mathbf{tr}(Z^{\frac{1}{2}}(X_0 - X_*)Z^{\frac{1}{2}})]^2 \\ &= (1/\lambda_n)[\mathbf{tr}((X_0 - X_*)Z)]^2 \\ &= (1/\lambda_n)[(X_0 - X_*) \bullet Z]^2,\end{aligned}$$

where we use the fact that for symmetric matrix $M \geq 0$, $\|M\|_F \leq \mathbf{tr}(M)$. Similarly,

$$\begin{aligned}\|D\mathbf{vec}(Z_0 - Z_*)\|^2 &= (\mathbf{vec}(Z_0 - Z_*))^T (E^{-1}SE^{-1})\mathbf{vec}(Z_0 - Z_*) \\ &= (\mathbf{vec}(Z_0 - Z_*))^T (E^{-1}F)\mathbf{vec}(Z_0 - Z_*) \\ &= (1/2)\mathbf{vec}(X^{\frac{1}{2}}(Z_0 - Z_*)X^{\frac{1}{2}})^T \\ &\quad (X^{-\frac{1}{2}}Z^{-1}X^{-\frac{1}{2}} \otimes I + I \otimes X^{-\frac{1}{2}}Z^{-1}X^{-\frac{1}{2}}) \\ &\quad \mathbf{vec}(X^{\frac{1}{2}}(Z_0 - Z_*)X^{\frac{1}{2}}) \\ &\leq (1/\lambda_n)\|\mathbf{vec}X^{\frac{1}{2}}(Z_0 - Z_*)X^{\frac{1}{2}}\|^2 \\ &= (1/\lambda_n)\|X^{\frac{1}{2}}(Z_0 - Z_*)X^{\frac{1}{2}}\|_F^2 \\ &\leq (1/\lambda_n)[\mathbf{tr}(X^{\frac{1}{2}}(Z_0 - Z_*)X^{\frac{1}{2}})]^2 \\ &= (1/\lambda_n)[\mathbf{tr}(X(Z_0 - Z_*))]^2 \\ &= (1/\lambda_n)[X \bullet (Z_0 - Z_*)]^2.\end{aligned}$$

Therefore,

$$(3.21) \quad \xi \leq \sqrt{1/\lambda_n} \nu[X \bullet (Z_0 - Z_*) + (X_0 - X_*) \bullet Z] = \sqrt{1/\lambda_n} \zeta,$$

where

$$\zeta = \nu[X \bullet (Z_0 - Z_*) + (X_0 - X_*) \bullet Z].$$

To estimate ζ , we invoke (3.11) to get

$$\begin{aligned}\zeta &\leq X \bullet Z + \nu X_0 \bullet Z_0 + \nu(X_0 \bullet Z_* + X_* \bullet Z_0) \\ &\leq n\mu + (\mu/\mu_0)n\mu_0 + (n\mu)/(n\mu_0)(X_0 \bullet Z_* + X_* \bullet Z_0) \\ &\leq n\mu[2 + (X_0 \bullet Z_* + X_* \bullet Z_0)/X_0 \bullet Z_0].\end{aligned}$$

Substituting the above into (3.21), we obtain

$$\xi \leq \sqrt{1/\lambda_n} n\mu \left(2 + \frac{X_0 \bullet Z_* + X_* \bullet Z_0}{X_0 \bullet Z_0} \right).$$

This proves the first inequality.

Under condition (3.19), we calculate that

$$\frac{X_0 \bullet Z_* + X_* \bullet Z_0}{X_0 \bullet Z_0} = \frac{\rho[\text{tr}(X_*) + \text{tr}(Z_*)]}{\rho^2 n} = \frac{\text{tr}(X_*) + \text{tr}(Z_*)}{\rho n} \leq 1.$$

Therefore,

$$\xi \leq 3n\mu\sqrt{1/\lambda_n}.$$

Finally, using the condition $\lambda_n \geq \gamma\mu$, we complete the proof. $\square$

LEMMA 3.8. *Under conditions (3.17), (3.18), and (3.19), if $\lambda_n \geq \gamma\mu$ for $\gamma \in (0, 1)$, then for $n \geq 4$,*

$$(3.22) \quad \|D^{-T} \text{vec} \Delta X\|^2 + \|D \text{vec} \Delta Z\|^2 \leq 38n^2 \mu / \gamma.$$

Proof. Observe that when $n \geq 4$, by Lemmas 3.6 and 3.7,

$$\xi^2 + \eta \leq 10n^2 \mu / \gamma.$$

Therefore, by Lemma 3.5,

$$\|D^{-T} \text{vec} \Delta X\|^2 + \|D \text{vec} \Delta Z\|^2 \leq \left(\xi + \sqrt{\xi^2 + \eta} \right)^2 \leq (3 + \sqrt{10})^2 n^2 \mu / \gamma \leq 38n^2 \mu / \gamma,$$

which completes the proof. $\square$

Under slightly different conditions and using some variants of Lemma 3.4, Kojima, Shindoh, and Hara [11] derived their Lemma 7.6, which has similar functionality as our Lemma 3.8. Our straightforward approach leads to a shorter derivation.

4. Central path and its neighborhoods. We first need to introduce some more notation. Let $\phi(M) \in \mathbf{C}^n$ be the vector of eigenvalues of the matrix $M \in \mathbf{R}^{n \times n}$, i.e., the elements of $\phi(M)$ form the spectrum of M . For convenience, if all eigenvalues are real we will assume the order

$$\phi_1(M) \geq \phi_2(M) \geq \cdots \geq \phi_{n-1}(M) \geq \phi_n(M).$$

We also define

$$X(\alpha) = X + \alpha \Delta X, \quad Z(\alpha) = Z + \alpha \Delta Z, \quad \mu(\alpha) = X(\alpha) \bullet Z(\alpha) / n.$$

4.1. Variable-metric measures of centrality. The following simple fact is somewhat obvious. We include a proof for completeness.

PROPOSITION 4.1. *For $M \in \mathbf{R}^{n \times n}$ with a real spectrum, nonsingular $P \in \mathbf{R}^{n \times n}$, and a scalar τ ,*

$$H_P(M) = \tau I \Leftrightarrow M = \tau I.$$

Proof. Suppose that the equality holds on the left. We must have

$$PMP^{-1} = \tau I + G$$

for some skew-symmetric matrix G . Since the spectrum of M is real, we must have $G = 0$, otherwise M would have an eigenvalue $\tau + \phi_k(G)$ for a nonzero pure imaginary number $\phi_k(G)$, contradicting the realness assumption for the spectrum of M . The converse is obvious. $\square$

Hence the central path $XZ = \mu I$ is equivalent to $H_P(XZ) = \mu I$ for any nonsingular $P \in \mathbf{R}^{n \times n}$. In particular, at a given point (X, Z) and for $P = Z^{\frac{1}{2}}$, the central path can be defined, for variable (X_+, Z_+) , as

$$(4.1) \quad H(X_+Z_+) = \mu_+ I \equiv \frac{\text{tr}H(X_+Z_+)}{n} I.$$

Moreover, the centrality of $(X(\alpha), Z(\alpha))$ can be measured by the magnitude of the quantity

$$(4.2) \quad H(X(\alpha)Z(\alpha)) - \mu(\alpha)I \equiv H(X(\alpha)Z(\alpha)) - \frac{\text{tr}H(X(\alpha)Z(\alpha))}{n} I.$$

It should be noted though that in (4.1) the scaling matrix $Z^{\frac{1}{2}}$ changes from iteration to iteration, making (4.1) a variable-metric definition of the central path and (4.2) a variable-metric measure of centrality.

It is extremely convenient, from a theoretical point of view, to use the new definition (4.1) for the central path in primal-dual methods for SDP because of its connection with the linearized complementarity (2.10), as is indicated by the following proposition obtained by direct calculation.

LEMMA 4.1. *Let (X, y, Z) and $(\Delta X, \Delta y, \Delta Z)$ satisfy the linear system (2.16). Then*

$$\begin{aligned} H(X(\alpha)Z(\alpha)) &= (1 - \alpha)H(XZ) + \alpha\sigma\mu I + \alpha^2 H(\Delta X \Delta Z), \\ \mu(\alpha) &= (1 - \alpha + \alpha\sigma)\mu + \frac{\alpha^2}{n} \Delta X \bullet \Delta Z. \end{aligned}$$

Corresponding to the two most popular centrality conditions in LP,

$$\|x \circ z - \mu e\| \leq \beta\mu \quad \text{and} \quad \gamma\mu \leq x \circ z \leq \Gamma\mu,$$

we have for SDP

$$(4.3) \quad \|H(XZ) - \mu I\|_F \leq \beta\mu, \quad \beta \in (0, 1),$$

and

$$(4.4) \quad \gamma\mu \leq \phi(H(XZ)) \leq \Gamma\mu,$$

where Γ can be infinity, in which case the condition is one-sided. More precisely, these conditions mean that the steplength α should satisfy

$$(4.5) \quad \|H(X(\alpha)Z(\alpha)) - \mu(\alpha)I\|_F \leq \beta_+ \mu(\alpha), \quad \beta_+ \in (0, 1),$$

or

$$(4.6) \quad \gamma_+ \mu(\alpha) \leq \phi(H(X(\alpha)Z(\alpha))) \leq \Gamma_+ \mu(\alpha),$$

where the values of the parameters β_+ , γ_+ , and Γ_+ may or may not be different from those of β , γ , and Γ at the previous iteration.

For a fixed steplength parameter $\alpha = \alpha_+$, let us define

$$X_+ = X(\alpha_+), \quad Z_+ = Z(\alpha_+), \quad \mu_+ = \mu(\alpha_+).$$

If Z_+ is s.p.d. (see Lemma 4.7 later in this section), then $Z_+^{\frac{1}{2}}$ will be the next scaling matrix P in (2.7) to be used in the next step. We will denote the operator H_P corresponding to $P = Z_+^{\frac{1}{2}}$ by H_+ . In general,

$$H(X_+Z_+) \neq H_+(X_+Z_+) \equiv Z_+^{\frac{1}{2}}X_+Z_+^{\frac{1}{2}},$$

though we do have, as observed by Monteiro in [15], for $P = Z_+^{\frac{1}{2}}Z_+^{-\frac{1}{2}}$,

$$(4.7) \quad H(X_+Z_+) = H_P(H_+(X_+Z_+)).$$

The following lemma guarantees a smooth transition from one iteration to the next for the variable-metric centrality conditions (4.5) and (4.6).

LEMMA 4.2. *Assume that both Z and Z_+ are s.p.d. Then*

1. $\|H_+(X_+Z_+) - \mu_+I\|_F \leq \|H(X_+Z_+) - \mu_+I\|_F$;
2. $\phi_n(H(X_+Z_+)) \leq \phi(H_+(X_+Z_+)) \equiv \phi(X_+Z_+) \leq \phi_1(H(X_+Z_+))$.

Proof. The first inequality was proved by Monteiro in [15] (see equation (32) in the proof of Theorem 4.1). The second set of inequalities follows from the fact that the real part of the spectrum of a real matrix is contained between the largest and the smallest eigenvalues of its Hermitian part (see page 187 of [6], for example). $\square$

We point out that although Monteiro [15] did not explicitly use the variable-metric central path (4.1), nor the variable-metric centrality conditions (4.5) and (4.6), he made the necessary connections, using the relations in Lemma 4.2, between the variable-metric central path definition (4.1) and the more traditional central path definition $\phi(XZ) = \mu e$ that he used.

4.2. Results for steplength selections. Subtracting the second equation times identity from the first one in Lemma 4.1, we obtain

$$(4.8) \quad H(X(\alpha)Z(\alpha)) - \mu(\alpha)I = (1-\alpha)[H(XZ) - \mu I] + \alpha^2 \left[H(\Delta X \Delta Z) - \frac{\Delta X \bullet \Delta Z}{n} I \right].$$

In order to estimate terms like the second term in the right-hand side of (4.8), we define

$$(4.9) \quad \omega = \|H(\Delta X \Delta Z)\|_F + \frac{2}{\sqrt{n}} |\Delta X \bullet \Delta Z|.$$

LEMMA 4.3. *Let $\chi = 0$ if (X, y, Z) is feasible and $\chi = 1$ otherwise. Then*

$$\omega \leq \left(\frac{\sqrt{\kappa}}{2} + \frac{\chi}{\sqrt{n}} \right) (\|D^{-T} \mathbf{vec} \Delta X\|^2 + \|D \mathbf{vec} \Delta Z\|^2).$$

Proof. If (X, y, Z) is feasible, then $\Delta X \bullet \Delta Z = 0$ and the lemma follows from Lemma 3.3. If (X, y, Z) is infeasible, then the lemma follows from the fact that

$$(4.10) \quad |\Delta X \bullet \Delta Z| \leq \|D^{-T} \mathbf{vec} \Delta X\| \|D \mathbf{vec} \Delta Z\| \leq \frac{1}{2} (\|D^{-T} \mathbf{vec} \Delta X\|^2 + \|D \mathbf{vec} \Delta Z\|^2).$$

This completes the proof. $\square$

From (4.10) and Lemma 3.8, we obtain the estimate

$$(4.11) \quad |\Delta X \bullet \Delta Z| \leq \frac{19n^2\mu}{\gamma},$$

which will be useful later.

In the rest of the paper, we demonstrate that the use of (4.5) and (4.6) enables us to obtain explicit rules for steplength selections almost exactly identical to those in LP. This fact greatly facilitates the extensions of many, if not most, primal-dual algorithms from LP to SDP. The lemmas in this subsection are all straightforward extensions of well-known steplength selection rules in LP.

For the strong centrality condition (4.3), which usually leads to the so-called short-step algorithms, the following result is useful, especially for variants of the Mizuno-Todd-Ye predictor-corrector algorithm [14] (which should not be classified as a short-step algorithm, though).

LEMMA 4.4. *Let (X, Z) satisfy (4.3). Then (4.5) holds for all $\alpha \in [0, 1]$ satisfying*

$$(4.12) \quad h(\alpha) = (\beta - \beta_+)(1 - \alpha)\mu - \beta_+\sigma\alpha\mu + \omega\alpha^2 \leq 0.$$

Moreover, (4.5) holds when

1. $\sigma = 0$, $\beta_+ > \beta$ and

$$(4.13) \quad \alpha \leq \frac{2}{1 + \sqrt{1 + 4\omega/(\beta_+ - \beta)\mu}};$$

2. (X, y, Z) is feasible, $\sigma = \alpha = 1$, and

$$(4.14) \quad \frac{\beta^2(1 + \beta)^{1/2}}{2(1 - \beta)^{3/2}} \leq \beta_+.$$

Proof. It follows from (4.8) and (4.3) that

$$\begin{aligned} & \|H(X(\alpha)Z(\alpha)) - \mu(\alpha)I\|_F - \beta_+\mu(\alpha) \\ & \leq (1 - \alpha)\beta\mu + \alpha^2(\|H(\Delta X \Delta Z)\|_F + |\Delta X \bullet \Delta Z|/\sqrt{n}) \\ & \quad - \beta_+((1 - \alpha + \alpha\sigma)\mu + \alpha^2\Delta X \bullet \Delta Z/n) \\ & \leq (\beta - \beta_+)(1 - \alpha)\mu - \beta_+\sigma\alpha\mu \\ & \quad + \alpha^2(\|H(\Delta X \Delta Z)\|_F + (1/\sqrt{n} + \beta_+/n)|\Delta X \bullet \Delta Z|) \\ & \leq h(\alpha). \end{aligned}$$

Therefore, $h(\alpha) \leq 0$ implies (4.5). When $\sigma = 0$ and $\beta_+ > \beta$, the right-hand side of (4.13) is the unique positive root of $h(\alpha)$. Hence (4.13) is equivalent to $h(\alpha) \leq 0$.

For the second case, observe that when $\sigma = \alpha = 1$, $h(\alpha) \leq 0$ is equivalent to

$$(4.15) \quad \omega \leq \beta_+ \mu.$$

Condition (4.3) implies

$$\|H(XZ) - \mu I\|_F = \|\Lambda - \mu I\|_F \leq \beta \mu,$$

and hence

$$(1 - \beta)\mu \leq \lambda_i \leq (1 + \beta)\mu.$$

When (X, Z) is feasible, from $\Delta X \bullet \Delta Z = 0$ and Corollary 3.1 we have

$$\omega \leq \frac{\sqrt{\kappa}}{2} \frac{\|\Lambda - \mu I\|_F^2}{\gamma \mu} \leq \frac{\sqrt{\kappa}}{2} \frac{\beta^2 \mu^2}{\gamma \mu} = \frac{\beta^2 \sqrt{\kappa}}{2\gamma} \mu$$

with $\gamma \geq 1 - \beta$ and $\kappa \leq (1 + \beta)/(1 - \beta)$. Substituting these bounds into (4.15), we obtain (4.14). $\square$

It is easy to check that the inequality (4.14) is satisfied by

$$\beta = 0.45 \quad \text{and} \quad \beta_+ = 0.3.$$

These values will be used later in section 6.3.

We are actually more interested in the weak centrality condition (4.6), which usually leads to the so-called long-step algorithms that are much more efficient in practice.

The following lemma gives an inequality for finding lower bounds on the steplength parameter when the weak centrality condition (4.6) is in use. It is worth observing that no eigenvalue calculations are necessary in order to find the lower bound.

LEMMA 4.5. *Let $\phi(H(XZ))$ satisfy (4.4) and $0 < \gamma_+ \leq \gamma < 1 < \Gamma \leq \Gamma_+$. Then the inequalities*

$$(4.16) \quad g(\alpha) = (1 - \alpha)(\gamma - \gamma_+)\mu + \alpha\sigma(1 - \gamma_+)\mu - \omega\alpha^2 \geq 0,$$

$$(4.17) \quad G(\alpha) = (1 - \alpha)(\Gamma_+ - \Gamma)\mu + \alpha\sigma(\Gamma_+ - 1)\mu - \omega\alpha^2 \geq 0,$$

imply, respectively,

$$\phi_n(H(X(\alpha)Z(\alpha))) \geq \gamma_+\mu(\alpha) \quad \text{and} \quad \phi_1(H(X(\alpha)Z(\alpha))) \leq \Gamma_+\mu(\alpha).$$

Moreover, centrality condition (4.6) holds for all $\alpha \in [0, 1]$ satisfying

$$\alpha \leq \min(1 - \gamma_+, \Gamma_+ - 1) \frac{\sigma\mu}{\omega}.$$

Proof. Using Lemma 4.1, the definition (4.9) for ω , and condition (4.4), we calculate

$$\phi_n(H(X(\alpha)Z(\alpha))) - \gamma_+\mu(\alpha) \geq (1 - \alpha)[\phi_n(H(XZ)) - \gamma\mu] + g(\alpha) \geq g(\alpha).$$

Hence, $g(\alpha) \geq 0$ implies $\phi_n(H(X(\alpha)Z(\alpha))) \geq \gamma_+\mu(\alpha)$. Similarly, we can derive that $G(\alpha) \geq 0$ implies $\phi_1(H(X(\alpha)Z(\alpha))) \leq \Gamma_+\mu(\alpha)$. Moreover, ignoring the first term, which is nonnegative, in $g(\alpha)$ we see that $\alpha \leq (1 - \gamma_+)\sigma\mu/\omega$ is sufficient for

$g(\alpha) \geq 0$. Similarly, $\alpha \leq (\Gamma_+ - 1)\sigma\mu/\omega$ is sufficient for $G(\alpha) \geq 0$. This completes the proof. $\square$

For infeasible-interior-point algorithms, we need to impose condition (3.17) on the steplength parameter α . Since ν is updated by a factor of $(1 - \alpha)$ at each iteration, in terms of α , condition (3.17) can be written as

$$(4.18) \quad \mu(\alpha) \geq (1 - \alpha)\mu.$$

It is worth noting that this relationship always holds for feasible algorithms.

LEMMA 4.6. *Condition (4.18) is satisfied for all positive α if $\Delta X \bullet \Delta Z \geq 0$; otherwise under the conditions of Lemma 3.8 it is satisfied for $\alpha \in [0, 1]$ such that*

$$\alpha \leq \frac{\sigma n \mu}{|\Delta X \bullet \Delta Z|} \leq \frac{\sigma \gamma}{19n}.$$

Proof. Condition (4.18) is equivalent to

$$\mu(\alpha) - (1 - \alpha)\mu = \alpha\sigma\mu + \frac{\alpha^2}{n}\Delta X \bullet \Delta Z \geq 0,$$

which always holds if $\Delta X \bullet \Delta Z \geq 0$. If $\Delta X \bullet \Delta Z < 0$, then condition (4.18) is equivalent to $\alpha \leq n\sigma\mu/|\Delta X \bullet \Delta Z|$, which, together with (4.11), proves the lemma. $\square$

4.3. Positive definiteness of X and Z . Finally, we need to ensure the positive definiteness of the matrices X and Z at each iteration. Specifically, we need to show that X_+ and Z_+ are s.p.d.

LEMMA 4.7. *Let $\alpha_+ \in [0, 1]$ be the largest value such that for all $\alpha \in [0, \alpha_+]$ condition (4.18), and either condition (4.5) or (4.6), are satisfied. Then both X_+ and Z_+ are s.p.d. unless $\alpha_+ = 1$ and $\mu_+ = 0$ in which case (X_+, y_+, Z_+) is a solution to (1.2).*

Proof. It suffices to verify for the weaker condition (4.6). We recall that the matrices X and Z are s.p.d. and note that condition (4.18) implies $\mu(\alpha) > 0$ for $\alpha \in [0, 1]$. If $\alpha_+ = 1$ and $\mu(1) = 0$, then (X_+, y_+, Z_+) is clearly a solution. So we assume that $\mu(\alpha) > 0$ for $\alpha \in [0, \alpha_+]$. Since the eigenvalues of $H(X(\alpha)Z(\alpha))$ are continuous with respect to α and $H(X(0)Z(0))$ is s.p.d., it follows from (4.6) and continuity that for $\alpha \in [0, \alpha_+]$ all the eigenvalues of $H(X(\alpha)Z(\alpha))$ remain positive. Hence by Lemma 4.2 all of those of $Z^{\frac{1}{2}}X(\alpha)Z(\alpha)Z^{-\frac{1}{2}}$, and in turn those of $X(\alpha)Z(\alpha)$, remain positive since they are bounded below by the smallest eigenvalue of the Hermitian part $H(X(\alpha)Z(\alpha))$. This implies that all the eigenvalues of both $X(\alpha)$ and $Z(\alpha)$ remain positive and completes the proof. $\square$

5. Condition Number of $Z^{\frac{1}{2}}XZ^{\frac{1}{2}}$. Most bounds derived in this paper depend on κ — the condition number of the matrix $Z^{\frac{1}{2}}XZ^{\frac{1}{2}}$, whose order in turn depends on how close XZ is to the central path. We include some well-known facts concerning κ below as a proposition for the sake of convenience.

PROPOSITION 5.1. *If XZ satisfies condition (4.3) for $\beta < 1$, then*

$$\kappa \leq \frac{1 + \beta}{1 - \beta}.$$

On the other hand, if XZ satisfies condition (4.4), then

$$\kappa \leq \frac{\min(n, \Gamma)}{\gamma}.$$

In general, for short-step algorithms using the centrality condition (4.3), κ , being $O(1)$, will not contribute to the order of iteration-complexity. However, for algorithms using the centrality condition (4.4), Γ has to be chosen as a constant independent of n in order to preserve the iteration-complexity order. When using the popular one-sided weak centrality condition, in which $\Gamma = \infty$ such as in the Kojima–Mizuno–Yoshise feasible path-following algorithm [10] and in Zhang’s infeasible-interior-point algorithm [25], we have $\kappa = O(n)$, thus increasing the iteration-complexity order.

6. Examples of extensions. To demonstrate the power of the results obtained in the previous sections, in this section we give some examples of extending polynomiality results to SDP. Among them are the Kojima–Mizuno–Yoshise path-following algorithm [10], the infeasible-interior-point algorithm of Zhang [25], both long-step algorithms using the one-sided weak centrality condition (4.4), and the Mizuno–Todd–Ye predictor-corrector algorithm [14]. The Kojima–Mizuno–Yoshise algorithm has been extended to SDP by Monteiro [15] and the Mizuno–Todd–Ye algorithm by Nesterov and Todd [17].

We will describe the algorithms and then give their complexity results for SDP. Since we have obtained the essential ingredients for complexity analysis that are in forms similar to those in LP, the proofs naturally bears much similarity to their counterparts in LP. In the rest of the paper, we will always assume that $\epsilon \in (0, 1)$.

6.1. Feasible path-following algorithms.

Algorithm-A:

Choose a strictly feasible point (X, y, Z) , $\beta, \sigma \in (0, 1)$, $0 < \gamma < 1 < \Gamma$, independent of n , such that either condition (4.3) or condition (4.4) is satisfied. Let $\mu_0 = X \bullet Z/n$.

Repeat until $\mu \leq \epsilon \mu_0$, **do**

- (1) Find a solution $(\Delta X, \Delta y, \Delta Z)$ to (2.16).
- (2) Choose the largest $\alpha_+ \leq 1$ such that either (4.5) or (4.6) is satisfied by all $\alpha \in [0, \alpha_+]$.
- (3) $(X, y, Z) \leftarrow (X, y, Z) + \alpha_+(\Delta X, \Delta y, \Delta Z)$.
- (4) $\mu = X \bullet Z/n$.

End

In the algorithm, either (4.5) or (4.6) is consistently used throughout the process. When (4.6) is used with $\Gamma = \infty$, the algorithm is the Kojima–Mizuno–Yoshise long-step path-following algorithm [10]—the first long-step polynomial primal–dual feasible-interior-point algorithm, which has recently been extended to SDP by Monteiro in [15].

PROPOSITION 6.1. *Assume that a strictly feasible point (X, y, Z) to (1.2) exists. Algorithm-A terminates, asymptotically, in at most $O(n^p \ln \frac{1}{\epsilon})$ iterations, where*

$$p = \begin{cases} 1, & \text{if (4.5) is used,} \\ 1, & \text{if (4.6) is used and } \Gamma < \infty, \\ 1.5, & \text{if (4.6) is used and } \Gamma = \infty. \end{cases}$$

Since the proof of Proposition 6.1 is similar to, and simpler than, that of Proposition 6.2 below, we omit it here.

6.2. Infeasible path-following algorithms.

Algorithm-B:

Choose (X, y, Z) satisfying (3.17)–(3.19), $\beta, \sigma \in (0, 1)$, $0 < \gamma < 1 < \Gamma$, independent of n , such that either condition (4.3) or condition (4.4) is satisfied. Let $\mu_0 = X \bullet Z/n$ and $\nu = 1$.

Repeat until $\mu \leq \epsilon\mu_0$, **do**

- (1) Find a solution $(\Delta X, \Delta y, \Delta Z)$ to (2.16).
- (2) Calculate the largest $\hat{\alpha} \leq 1$ such that (4.18), and either (4.5) or (4.6), are satisfied by all $\alpha \in [0, \hat{\alpha}]$.
- (3) Choose α_+ to minimize $\mu(\alpha) = X(\alpha) \bullet Z(\alpha)/n$ in $[0, \hat{\alpha}]$.
- (4) $(X, y, Z) \leftarrow (X, y, Z) + \alpha_+(\Delta X, \Delta y, \Delta Z)$.
- (5) $\mu = X \bullet Z/n$ and $\nu \leftarrow (1 - \alpha_+)\nu$.

End

In the algorithm, either (4.5) or (4.6) is consistently used throughout the process. When (4.6) is used with $\Gamma = \infty$, the algorithm is an extension of the infeasible-interior-point algorithm studied in [25]. We note that in steps (2) and (3) it is not necessary to pick the exact maximizer $\hat{\alpha}$ and minimizer α_+ .

We observe that because of condition (4.18), the decrease of infeasibility is never slower than that of duality gap. Hence a sufficiently small duality gap will ensure a small infeasibility, justifying the termination condition in Algorithm-B.

PROPOSITION 6.2. *Assume that a solution to (1.2) exists. Algorithm-B terminates, asymptotically, in at most $O(n^p \ln \frac{1}{\epsilon})$ iterations, where*

$$p = \begin{cases} 2, & \text{if (4.5) is used,} \\ 2, & \text{if (4.6) is used and } \Gamma < \infty, \\ 2.5, & \text{if (4.6) is used and } \Gamma = \infty. \end{cases}$$

Proof. We will prove the proposition for condition (4.6) only since the proof for condition (4.5) is very similar except that (4.12) in Lemma 4.4 would be used instead of Lemma 4.5. We note that Step (1) in Algorithm-B is well defined by the virtue of Proposition 2.2.

Consider Step (2) in Algorithm-B. It follows from Lemmas 4.5 and 4.6 that $\hat{\alpha}$ in Step (2) exists and satisfies

$$\hat{\alpha} \geq \min \left\{ \min(1 - \gamma, \Gamma - 1) \frac{\sigma\mu}{\omega}, \frac{\sigma\gamma}{19n} \right\}.$$

Furthermore, Lemmas 3.8 and 4.3 ensure the existence of a constant β_1 such that

$$(6.1) \quad \hat{\alpha} \geq \bar{\alpha} \equiv \frac{\beta_1}{\sqrt{\kappa}n^2}, \quad \beta_1 \in \left(0, \frac{(1 - \sigma)\gamma n}{19} \right).$$

The upper bound on β_1 will be utilized soon.

Now consider Step (3) in Algorithm-B. Observe from the expression of $\mu(\alpha)$ in Lemma 4.1 that $\alpha_+ = \hat{\alpha}$ if $\Delta X \bullet \Delta Z \leq 0$. On the other hand, if $\Delta X \bullet \Delta Z > 0$, then

$$\alpha_+ = \min \left\{ \hat{\alpha}, \frac{(1 - \sigma)n\mu}{2\Delta X \bullet \Delta Z} \right\}.$$

Since α_+ minimizes $\mu(\alpha)$ in $[0, \hat{\alpha}]$ and $\bar{\alpha} \in [0, \hat{\alpha}]$, clearly, $\mu(\alpha_+) \leq \mu(\bar{\alpha})$. At any two consecutive iterations, invoking (4.11) and substituting $\bar{\alpha} = \beta_1/(\sqrt{\kappa}n^2)$ we have

$$\begin{aligned} \mu(\alpha_+)/\mu &\leq \mu(\bar{\alpha})/\mu = 1 - \bar{\alpha}[1 - \sigma - \bar{\alpha}\Delta X \bullet \Delta Z/(n\mu)] \\ &\leq 1 - \bar{\alpha}[1 - \sigma - |\Delta X \bullet \Delta Z|\beta_1/(\sqrt{\kappa}n^3\mu)] \\ &\leq 1 - \bar{\alpha}[1 - \sigma - 19\beta_1/(\gamma\sqrt{\kappa}n)] \\ &\leq 1 - [\beta_1/(\sqrt{\kappa}n^2)][1 - \sigma - 19\beta_1/(\gamma n)] \\ &= 1 - \beta_2/(\sqrt{\kappa}n^2), \end{aligned}$$

where $\beta_2 = \beta_1[1 - \sigma - 19\beta_1/(\gamma n)] > 0$ in view of the upper bound on β_1 in (6.1). This means that the duality gap is reduced at each iteration by at least a factor of $1 - \beta_2/(\sqrt{\kappa}n^2)$. By a standard argument, we conclude that $\mu \leq \epsilon\mu_0$ in at most $O(\sqrt{\kappa}n^2 \ln \frac{1}{\epsilon})$ iterations.

Finally, the iteration-complexity orders for the cases $\Gamma < \infty$ and $\Gamma = \infty$ both follow from Proposition 5.1 and the discussion after it. $\square$

Infeasible-interior-point algorithms require weaker conditions to operate and they are usually more efficient in practice. However, their theoretical complexities are higher than their counterparts in feasible-interior-point algorithms.

6.3. Mizuno–Todd–Ye predictor-corrector algorithm.

Algorithm-C:

Choose a strictly feasible point (X, y, Z) such that condition (4.3) is satisfied with $\beta = 0.3$. Let $\mu_0 = X \bullet Z/n$.

Repeat until $\mu \leq \epsilon\mu_0$, **do**

- (1) Find a solution $(\Delta X, \Delta y, \Delta Z)$ to (2.16) for $\sigma = 0$.
- (2) Choose the largest $\alpha_+ \leq 1$ such that (4.5) is satisfied with $\beta_+ = 0.45$ by all $\alpha \in [0, \alpha_+]$.
- (3) $(X, y, Z) \leftarrow (X, y, Z) + \alpha_+(\Delta X, \Delta y, \Delta Z)$. $\mu = X \bullet Z/n$.
- (4) Find a solution $(\Delta X, \Delta y, \Delta Z)$ to (2.16) for $\sigma = 1$.
- (5) $(X, y, Z) \leftarrow (X, y, Z) + (\Delta X, \Delta y, \Delta Z)$.

End

At each iteration, the algorithm solves two linear systems to compute two different steps: the predictor-step computed in (1) and the corrector-step in (4).

Using a different linearized complementarity equation, Nesterov and Todd [17] have recently extended the Mizuno–Todd–Ye predictor-corrector algorithm to SDP.

PROPOSITION 6.3. *Assume that a strictly feasible point (X, y, Z) to (1.2) exists. Algorithm-C terminates in at most $O(n^{\frac{1}{2}} \ln \frac{1}{\epsilon})$ iterations.*

Lemma 4.4 contains all the needed ingredients for the proof of Proposition 6.3, which is a direct translation from its counterparts for LP. We provide an outline of the proof in the next paragraph and refer the interested reader to [14] for more details.

Being a feasible algorithm, the goal of Algorithm-C is to drive the duality gap μ to zero. Observe that the predictor-step ($\sigma = 0$) decreases the duality gap by a factor of $1 - \alpha$ and the corrector-step ($\sigma = \alpha = 1$) does not alter the duality gap. At the beginning of any iteration, the iterate satisfies the centrality condition (4.3) with $\beta = 0.3$. First, (4.13) in Lemma 4.4 ensures that the step-length for the predictor-step is at least the order of $1/O(\sqrt{n})$ since

$$\beta_+ - \beta = 0.45 - 0.3 = 0.15, \quad \omega = O(n\mu), \quad \kappa \leq (1 + 0.3)/(1 - 0.3) \leq 2.$$

Second, (4.14) in Lemma 4.4 ensures that after the corrector-step the next iterate will again satisfy the centrality condition (4.3) with $\beta = 0.3$. Hence, the algorithm

guarantees a factor of $1 - 1/O(\sqrt{n})$ reduction in duality gap at every iteration, leading to the $O(\sqrt{n} \ln \frac{1}{\epsilon})$ -iteration complexity.

6.4. Other algorithms. A number of other primal-dual algorithms can be readily extended to SDP. For example, extensions of the Mizuno-Todd-Ye predictor-corrector algorithm to infeasible starting point should have an $O(n \ln \frac{1}{\epsilon})$ -iteration complexity. Recently, Potra and Sheng [20] and Kojima, Shida, and Shindoh [13] have independently extended infeasible interior-point predictor-corrector algorithms to SDP with $O(n \ln \frac{1}{\epsilon})$ -iteration complexity.

7. Concluding remarks. In summary, we consider that the present paper brings three additions to the research on primal-dual interior point methods for SDP.

1. We provided a concise derivation of the most basic equalities and inequalities for complexity analysis, almost completely parallel to the simple derivation for LP. Consequently, the derived results have familiar and highly compact forms.
2. We introduced variable-metric measures of centrality, making the task of extending polynomiality results to SDP much amenable for many existing primal-dual algorithms.
3. As an example, we extended a long-step infeasible-interior-point polynomial algorithm to SDP for the first time. This gives another theoretical indication that SDP, though more difficult than LP, is susceptible to practically more efficient algorithms.

We observe that for the infeasible algorithm, Algorithm-B, a stopping criterion can be derived from, among a number of possibilities, inequality (3.22) to detect “infeasibility.” Whenever (3.22) fails, we can safely conclude from Lemma 3.8 that there exists no solution (X_*, Z_*) satisfying the conditions (3.18) and (3.19). In particular, there is no solution (X_*, Z_*) satisfying

$$(X_*, Z_*) \leq \frac{1}{2} \rho(I, I),$$

where $\rho > 0$ and $X_0 = Z_0 = \rho I$.

Finally, we mention that results in this paper are applicable to more general problems such as the monotone linear complementarity problem in symmetric matrices studied by Kojima, Shindoh, and Hara in [11].

Appendix. We list some important operation rules for Kronecker products that are used in the present paper. We refer interested readers to Chapter 4 of the excellent book by Horn and Johnson [6]. We use $\phi(A)$ to denote the spectrums of A .

1. $A \otimes B = [a_{ij}B]$.
2. $\text{vec}(AXB) = (B^T \otimes A)\text{vec } X$.
3. $(A \otimes B)^T = A^T \otimes B^T$.
4. $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$.
5. $(A \otimes B)(C \otimes D) = AC \otimes BD$.
6. $\phi(A) = \{\mu_i\}$, $\phi(B) = \{\nu_i\} \Rightarrow \phi(A \otimes B) = \{\mu_i \nu_j\}$.

Acknowledgments. I sincerely thank the following colleagues: Renato Monteiro for pointing out to me a costly error in my first manuscript; Masakazu Kojima for bringing to my attention the equivalence of the search directions in [11] and [15]; and Steve Wright for his service in maintaining the interior-point mailing list and archive. I am also grateful to two anonymous referees for their constructive comments.

REFERENCES

- [1] F. ALIZADEH, *Interior point methods in semidefinite programming with applications to combinatorial optimization*, SIAM J. Optim., 5 (1995), pp. 13–51.
- [2] F. ALIZADEH, J. A. HAEGERLY, AND M. OVERTON, *Primal–Dual Interior-Point Methods for Semidefinite Programming: Convergence rates, stability and numerical results*, Tech. report 721, Courant Institute of Mathematical Sciences, New York University, New York, 1996.
- [3] F. ALIZADEH, J. A. HAEGERLY, AND M. OVERTON, *Complementarity and nondegeneracy in semidefinite programming*, Math. Programming, 77 (1997), pp. 111–128.
- [4] S. BOYD, L. GHOUI, E. FERON, AND V. BALAKRISHNAN, *Linear Matrix Inequalities in System and Control Theory*, SIAM, Philadelphia, PA, 1994.
- [5] C. HELMBERG, F. RENDL, R. J. VANDERBEI, AND H. WOLKOWICZ, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.
- [6] R. A. HORN AND C. R. JOHNSON, *Topics in Matrix Analysis*, Cambridge University Press, London, 1991.
- [7] F. JARRE, *An interior-point method for minimizing the maximum eigenvalue of a linear combination of matrices*, SIAM J. Control Optim., 31 (1993), pp. 1360–1377.
- [8] M. KOJIMA, *private communication*, Tokyo Institute of Technology, Japan, 1995.
- [9] M. KOJIMA, N. MEGIDDO, AND S. MIZUNO, *A primal–dual infeasible-interior-point algorithm for linear programming*, Math. Programming, 61 (1993), pp. 263–280.
- [10] M. KOJIMA, S. MIZUNO, AND A. YOSHISE, *A primal–dual interior point method for linear programming*, in Progress in mathematical programming, interior-point and related methods, Nimrod Megiddo, ed., Springer-Verlag, New York, 1989, pp. 29–47.
- [11] M. KOJIMA, S. SHINDOH, AND S. HARA, *Interior-point methods for the monotone linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.
- [12] M. KOJIMA, S. SHINDOH, AND S. HARA, *Reduction of monotone linear complementarity problems over cones to linear programs over cones*, Acta Math. Vietnam, 22 (1997), pp. 147–157.
- [13] M. KOJIMA, M. SHIDA, AND S. SHINDOH, *Local convergence of predictor–corrector infeasible-interior-point algorithms for SDPs and SDLCPs*, Math. Programming, 80 (1998), pp. 129–161.
- [14] S. MIZUNO, M. J. TODD, AND Y. YE, *On adaptive step primal–dual interior-point algorithms for linear programming*, Math. Oper. Res., 18 (1993), pp. 964–981.
- [15] R. C. MONTEIRO, *Primal–dual path following algorithms for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 663–678.
- [16] Y. E. NESTEROV AND A. S. NEMIROVSKII, *Interior Point Methods in Convex Programming—Theory and Applications*, SIAM, Philadelphia, PA, 1994.
- [17] Y. E. NESTEROV AND M. TODD, *Primal–dual interior-point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.
- [18] M. L. OVERTON, *On minimizing the maximum eigenvalue of a symmetric matrix*, SIAM J. Matrix Anal. Appl., 9 (1988), pp. 256–268.
- [19] M. L. OVERTON, *Large-scale optimization of eigenvalues*, SIAM J. Optim., 2 (1992), pp. 88–120.
- [20] F. A. POTRA AND RONGQIN SHENG, *A superlinearly convergent primal–dual infeasible-interior-point algorithm for semidefinite programming*, SIAM J. Optim., to appear.
- [21] M. RAMANA, *An exact duality theory for semidefinite programming and its complexity implications*, Math. Programming, 77 (1997), pp. 129–162.
- [22] F. RENDL, R. J. VANDERBEI, AND H. WOLKOWICZ, *Max-Min Eigenvalue Problems, primal-dual interior point algorithms, and trust subproblems*, Optim. Methods Softw., 5 (1995), pp. 1–16.
- [23] L. VANDENBERGHE AND S. BOYD, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.
- [24] L. VANDENBERGHE AND S. BOYD, *A primal–dual potential reduction method for problems involving matrix inequalities*, Math. Programming, Series B, 69 (1995), pp. 205–236.
- [25] YIN ZHANG, *On the convergence of a class of infeasible interior-point algorithms for the horizontal linear complementarity problem*, SIAM J. Optim., 4 (1994), pp. 208–227.

EXISTENCE AND UNIQUENESS OF SEARCH DIRECTIONS IN INTERIOR-POINT ALGORITHMS FOR THE SDP AND THE MONOTONE SDLCP*

MASAYUKI SHIDA[†], SUSUMU SHINDOH[‡], AND MASAKAZU KOJIMA[§]

Abstract. Various search directions used in interior-point algorithms for the semidefinite program (SDP) and the monotone semidefinite linear complementarity problem (SDLCP) are characterized by the intersection of a maximal monotone affine subspace and a maximal and strictly antitone affine subspace. This observation provides a unified geometric view over the existence of those search directions.

Key words. interior-point algorithm, semidefinite program, semidefinite linear complementarity problem, monotonicity

AMS subject classifications. 90C33, 65K05, 90C05, 90C25

PII. S1052623496300611

1. Introduction. The primal-dual interior-point method is known as one of the most efficient computational methods for solving large scale linear programs. There are numerous variations of the method, but most of them basically utilize a common search direction, which was introduced in the original primal-dual interior-point method [10, 18] and is often interpreted as the Newton direction towards a point on the central trajectory. By the primal-dual search direction, we mean this common search direction when we are concerned with interior-point methods for linear programs. The primal-dual search direction was extended naturally to the monotone linear complementarity problem. See [8, 9, etc.]. Several distinct search directions have been proposed, however, in recent primal-dual interior-point methods [1, 2, 6, 14, 16, 21, etc.] for the SDP and interior-point methods [12] for the monotone SDLCP. Many of them may be regarded as extensions of the primal-dual search direction used for linear programs, but they are essentially different.

The purpose of this short note is to give a characterization of those search directions proposed for the SDP and the monotone SDLCP in terms of the intersection of a maximal monotone affine subspace and a strictly and maximal antitone affine subspace. This characterization provides us with not only a simple proof of their unique existence but also a unified geometric view over those search directions. We deal with five kinds of search directions: one search direction from [16] for the SDP, one from [14] for the SDP, one from [2, 6] for the SDP, one from [12] for the monotone SDLCP, and one from [2] for the SDP.

In section 2, we state some basic definitions and general lemmas related to the concepts of monotonicity and antitonicity. In section 3, we present the main theorem (Theorem 3.1). Although we will restrict ourselves to the SDP and the monotone

*Received by the editors March 15, 1996; accepted for publication (in revised form) February 3, 1997.

<http://www.siam.org/journals/siopt/8-2/30061.html>

[†]Department of Mathematics, Kanagawa University, Rokkakubashi, Yokohama 221, Japan (shida@is.titech.ac.jp).

[‡]Department of Mathematics and Physics, The National Defense Academy, Hashirimizu, Yokosuka 239, Japan (shindoh@cc.nda.ac.jp).

[§]Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Oh-okayama, Meguro-ku, Tokyo 152, Japan (kojima@is.titech.ac.jp).

SDLCP, we can apply the lemmas and the characterization of search directions to other various problems (see remark (A) in section 4).

2. Basic lemmas. Let $\mathcal{H}$ be a finite dimensional real vector space with the inner product $\mathbf{x} \cdot \mathbf{y}$ of $\mathbf{x} \in \mathcal{H}$ and $\mathbf{y} \in \mathcal{H}$. We call a subset $\mathcal{A}$ of $\mathcal{H} \times \mathcal{H}$ *monotone* (or *antitone*, resp.) if $(\mathbf{u} - \mathbf{u}') \cdot (\mathbf{v} - \mathbf{v}') \geq 0$ (or ≤ 0 , resp.) for every pair $(\mathbf{u}, \mathbf{v}) \in \mathcal{A}$ and $(\mathbf{u}', \mathbf{v}') \in \mathcal{A}$. If the strict inequality holds for any distinct pair $(\mathbf{u}, \mathbf{v}) \in \mathcal{A}$ and $(\mathbf{u}', \mathbf{v}') \in \mathcal{A}$, we call $\mathcal{A}$ *strictly monotone* (or *strictly antitone*, resp.). We say that a monotone (or antitone, resp.) subset $\mathcal{A}$ is *maximal* if there is no monotone (or antitone, resp.) subset of $\mathcal{H} \times \mathcal{H}$ which properly contains $\mathcal{A}$. See [3, 5, 13, etc.].

We state two lemmas and one corollary. Lemma 2.1 and Corollary 2.2 give some characterizations of a maximal monotone (or maximal antitone) affine subspace of $\mathcal{H} \times \mathcal{H}$. Lemma 2.3 plays a key role in proving the existence and uniqueness of search directions in section 3.

LEMMA 2.1. *Let $\mathcal{A}$ be a monotone (or antitone, resp.) affine subspace of $\mathcal{H} \times \mathcal{H}$. Then $\mathcal{A}$ is maximal monotone (maximal antitone, resp.) if and only if $\dim \mathcal{A} = \dim \mathcal{H}$.*

Proof. Although the assertion of the lemma is well known (see, e.g., [5]), we include an elementary proof here. Since we know that an affine subspace $\mathcal{A}$ of $\mathcal{H} \times \mathcal{H}$ is (maximal) monotone if and only if $\{(\mathbf{x}, -\mathbf{y}) \in \mathcal{H} \times \mathcal{H} : (\mathbf{x}, \mathbf{y}) \in \mathcal{A}\}$ is (maximal) antitone, we only consider the case where $\mathcal{A}$ is a monotone affine subspace of $\mathcal{H} \times \mathcal{H}$. Let $m = \dim \mathcal{H}$. Choose an $(\mathbf{x}^0, \mathbf{y}^0) \in \mathcal{A}$, and let $\mathcal{A}_0 = \{(\mathbf{x} - \mathbf{x}^0, \mathbf{y} - \mathbf{y}^0) : (\mathbf{x}, \mathbf{y}) \in \mathcal{A}\}$. Then $\mathcal{A}_0$ is a monotone linear subspace of $\mathcal{H} \times \mathcal{H}$. It suffices to prove that $\mathcal{A}_0$ is maximal monotone if and only if $\dim \mathcal{A}_0 = m$. To prove the “if” part, let $\dim \mathcal{A}_0 = m$. Assume on the contrary that $\mathcal{A}_0$ is not maximal, i.e., there is a nonzero vector $(\bar{\mathbf{u}}, \bar{\mathbf{v}}) \in (\mathcal{H} \times \mathcal{H}) \setminus \mathcal{A}_0$ such that $\mathcal{A}_0 \cup \{(\bar{\mathbf{u}}, \bar{\mathbf{v}})\}$ is a monotone subset in $\mathcal{H} \times \mathcal{H}$. Then we have for every $(\mathbf{u}^{A_0}, \mathbf{v}^{A_0}) \in \mathcal{A}_0$,

$$(1) \quad (\mathbf{u}^{A_0} - \bar{\mathbf{u}}) \cdot (\mathbf{v}^{A_0} - \bar{\mathbf{v}}) \geq 0.$$

Let $\bar{\mathcal{A}} := \mathcal{A}_0 + \mathcal{R}(\bar{\mathbf{u}}, \bar{\mathbf{v}}) := \{(\mathbf{u}, \mathbf{v}) = (\mathbf{u}^{A_0}, \mathbf{v}^{A_0}) + r(\bar{\mathbf{u}}, \bar{\mathbf{v}}) : (\mathbf{u}^{A_0}, \mathbf{v}^{A_0}) \in \mathcal{A}_0, r \in \mathcal{R}\}$. Then $\bar{\mathcal{A}}$ is an $(m+1)$ -dimensional linear subspace of $\mathcal{H} \times \mathcal{H}$. We show that the linear subspace $\bar{\mathcal{A}}$ is monotone. Let $(\mathbf{u}, \mathbf{v}) = (\mathbf{u}^{A_0}, \mathbf{v}^{A_0}) + r(\bar{\mathbf{u}}, \bar{\mathbf{v}})$ be an arbitrary element of $\bar{\mathcal{A}}$. If $r = 0$, we have $\mathbf{u} \cdot \mathbf{v} = \mathbf{u}^{A_0} \cdot \mathbf{v}^{A_0} \geq 0$ since $\mathcal{A}_0$ is a monotone subspace. If $r \neq 0$, we have

$$\begin{aligned} \mathbf{u} \cdot \mathbf{v} &= (\mathbf{u}^{A_0} + r\bar{\mathbf{u}}) \cdot (\mathbf{v}^{A_0} + r\bar{\mathbf{v}}) = r^2 \left(-\frac{\mathbf{u}^{A_0}}{r} - \bar{\mathbf{u}} \right) \cdot \left(-\frac{\mathbf{v}^{A_0}}{r} - \bar{\mathbf{v}} \right) \\ &\geq 0 \quad \text{since } \mathcal{A}_0 \text{ is a linear subspace and inequality (1) holds.} \end{aligned}$$

Therefore, the $(m+1)$ -dimensional linear subspace $\bar{\mathcal{A}}$ is monotone. Let $\mathcal{B}_a = \{(\mathbf{w}, -\mathbf{w}) \in \mathcal{H} \times \mathcal{H} : \mathbf{w} \in \mathcal{H}\}$. It is easy to see that $\mathcal{B}_a$ is an m -dimensional strictly antitone linear subspace of $\mathcal{H} \times \mathcal{H}$. Since $\dim \bar{\mathcal{A}} + \dim \mathcal{B}_a = 2m + 1$, we have $\dim(\bar{\mathcal{A}} \cap \mathcal{B}_a) \geq 1$. This implies that there is a nonzero vector $(\mathbf{p}, -\mathbf{p}) \in \bar{\mathcal{A}} \cap \mathcal{B}_a$. This contradicts the fact that $\bar{\mathcal{A}}$ is monotone. To prove the “only if” part, we show that there is an m -dimensional monotone linear subspace $\mathcal{D}$ of $\mathcal{H} \times \mathcal{H}$ containing $\mathcal{A}_0$ if $\dim \mathcal{A}_0 < m$. Let $\mathcal{B}_m = \{(\mathbf{w}, \mathbf{w}) : \mathbf{w} \in \mathcal{H}\}$ and $\mathcal{D} = \mathcal{A}_0 + \mathcal{B}_m \cap \mathcal{A}_0^\perp$, where $\mathcal{C}^\perp$ denotes the orthogonal complement of a linear subspace $\mathcal{C}$ of $\mathcal{H} \times \mathcal{H}$. Then $\mathcal{B}_m$ is an m -dimensional strictly monotone linear subspace of $\mathcal{H} \times \mathcal{H}$ and $\mathcal{B}_m^\perp = \mathcal{B}_a$. Since $\mathcal{B}_a$ is strictly antitone and

$\mathcal{A}_0$ is monotone, we have $\mathcal{B}_a \cap \mathcal{A}_0 = \{\mathbf{0}\}$. Hence

$$\begin{aligned}\dim(\mathcal{B}_m \cap \mathcal{A}_0^\perp) &= \dim(\mathcal{B}_a + \mathcal{A}_0)^\perp = 2m - (m + \dim \mathcal{A}_0) = m - \dim \mathcal{A}_0, \\ \dim \mathcal{D} &= \dim(\mathcal{A}_0 + \mathcal{B}_m \cap \mathcal{A}_0^\perp) = \dim \mathcal{A}_0 + \dim(\mathcal{B}_m \cap \mathcal{A}_0^\perp) = m.\end{aligned}$$

Thus we have shown that $\dim \mathcal{D} = m$. To show the monotonicity of $\mathcal{D}$, suppose that $(\mathbf{u}, \mathbf{v}) \in \mathcal{D}$. Then there exist $(\mathbf{x}, \mathbf{y}) \in \mathcal{A}_0$ and $(\mathbf{w}, \mathbf{w}) \in \mathcal{B}_m \cap \mathcal{A}_0^\perp$ such that $(\mathbf{u}, \mathbf{v}) = (\mathbf{x} + \mathbf{w}, \mathbf{y} + \mathbf{w})$. It follows that

$$\begin{aligned}\mathbf{u} \cdot \mathbf{v} &= (\mathbf{x} + \mathbf{w}) \cdot (\mathbf{y} + \mathbf{w}) \\ &= \mathbf{x} \cdot \mathbf{y} + (\mathbf{x}, \mathbf{y}) \cdot (\mathbf{w}, \mathbf{w}) + \mathbf{w} \cdot \mathbf{w} \\ &= \mathbf{x} \cdot \mathbf{y} + \mathbf{w} \cdot \mathbf{w} \quad (\text{since } (\mathbf{x}, \mathbf{y}) \in \mathcal{A}_0 \text{ and } (\mathbf{w}, \mathbf{w}) \in \mathcal{A}_0^\perp) \\ &\geq 0 \quad (\text{since } \mathcal{A}_0 \text{ is monotone}). \quad \square\end{aligned}$$

COROLLARY 2.2. *Let $\mathcal{H}$ be a finite dimensional real vector space. Suppose that $\mathcal{A}$ is an affine subspace of $\mathcal{H} \times \mathcal{H}$ given by $\mathcal{A} = \{(\mathbf{x}, \mathbf{y}) \in \mathcal{H} \times \mathcal{H} : \mathbf{y} = \mathbf{Q}\mathbf{x} + \mathbf{q}, \mathbf{x} \in \mathcal{H}\}$, where $\mathbf{Q}$ is a linear operator on $\mathcal{H}$ and $\mathbf{q} \in \mathcal{H}$. Then $\mathcal{A}$ is strictly and maximal monotone (or strictly and maximal antitone, resp.) if and only if $\mathbf{Q}$ is positive definite; i.e., $\mathbf{u} \cdot \mathbf{Q}\mathbf{u} > 0$ for every nonzero $\mathbf{u} \in \mathcal{H}$ (or negative definite; i.e., $\mathbf{u} \cdot \mathbf{Q}\mathbf{u} < 0$ for every nonzero $\mathbf{u} \in \mathcal{H}$, resp.).*

Proof. From the definition of $\mathcal{A}$, we know that $\dim \mathcal{A} = \dim \mathcal{H}$ (the fact does not depend on whether the linear operator $\mathbf{Q}$ is positive definite or not). By Lemma 2.1, $\mathcal{A}$ is maximal monotone (or maximal antitone, resp.) whenever it is monotone (or antitone, resp.). On the other hand, it is easily verified that $\mathcal{A}$ is strictly monotone (or strictly antitone, resp.) if and only if $\mathbf{Q}$ is positive definite (or negative definite, resp.). Thus, we conclude the assertion. $\square$

LEMMA 2.3.

- (i) *Assume that an affine subspace $\mathcal{B}$ of $\mathcal{H} \times \mathcal{H}$ is strictly and maximal antitone. Then, $\mathcal{A} \cap \mathcal{B}$ is a singleton for every maximal monotone affine subspace $\mathcal{A}$ of $\mathcal{H} \times \mathcal{H}$.*
- (ii) *Assume that an affine subspace $\mathcal{B}$ of $\mathcal{H} \times \mathcal{H}$ does not contain a maximal and strictly antitone affine subspace. Then $\mathcal{A} \cap \mathcal{B} = \emptyset$ for some maximal monotone affine subspace $\mathcal{A}$ of $\mathcal{H} \times \mathcal{H}$.*

Proof.

- (i). Let $\mathcal{A}$ be a maximal monotone affine subspace of $\mathcal{H} \times \mathcal{H}$. If $\mathcal{A} \cap \mathcal{B}$ contains at least two points $(\mathbf{u}_1, \mathbf{v}_1), (\mathbf{u}_2, \mathbf{v}_2)$, we have

$$\begin{aligned}0 &\leq (\mathbf{u}_1 - \mathbf{u}_2) \cdot (\mathbf{v}_1 - \mathbf{v}_2) && \text{since } \mathcal{A} \text{ is monotone} \\ &< 0 && \text{since } \mathcal{B} \text{ is strictly antitone,}\end{aligned}$$

which is a contradiction. We show that $\mathcal{A} \cap \mathcal{B}$ is nonempty. Let

$$\begin{aligned}\mathcal{A}_0 &:= \mathcal{A} - (\mathbf{u}^A, \mathbf{v}^A) && \text{for some fixed } (\mathbf{u}^A, \mathbf{v}^A) \in \mathcal{A}, \\ \mathcal{B}_0 &:= \mathcal{B} - (\mathbf{u}^B, \mathbf{v}^B) && \text{for some fixed } (\mathbf{u}^B, \mathbf{v}^B) \in \mathcal{B}.\end{aligned}$$

Then $\mathcal{A}_0$ is a maximal monotone linear subspace of $\mathcal{H} \times \mathcal{H}$, and $\mathcal{B}_0$ is a strictly and maximal antitone linear subspace of $\mathcal{H} \times \mathcal{H}$. By the argument in the proof of Lemma 2.1, we have $\mathcal{A}_0 \oplus \mathcal{B}_0 = \mathcal{H} \times \mathcal{H}$. Hence, there exist $(\mathbf{x}^{A_0}, \mathbf{y}^{A_0}) \in \mathcal{A}_0$ and $(\mathbf{x}^{B_0}, \mathbf{y}^{B_0}) \in \mathcal{B}_0$ such that $(\mathbf{u}^A, \mathbf{v}^A) - (\mathbf{u}^B, \mathbf{v}^B) = (\mathbf{x}^{A_0}, \mathbf{y}^{A_0}) + (\mathbf{x}^{B_0}, \mathbf{y}^{B_0})$. Therefore, we have

$$\mathcal{A} \ni -(\mathbf{x}^{A_0}, \mathbf{y}^{A_0}) + (\mathbf{u}^A, \mathbf{v}^A) = (\mathbf{x}^{B_0}, \mathbf{y}^{B_0}) + (\mathbf{u}^B, \mathbf{v}^B) \in \mathcal{B};$$

thus we have $\mathcal{A} \cap \mathcal{B} \neq \emptyset$.

(ii). Let $\mathcal{B}_0 := \mathcal{B} - (\mathbf{u}^B, \mathbf{v}^B)$ for some fixed $(\mathbf{u}^B, \mathbf{v}^B) \in \mathcal{B}$. Let $\mathcal{C}$ be a maximal linear subspace of $\mathcal{B}_0$ with strict antitone property and $\mathcal{D}$ be its orthogonal complement in $\mathcal{B}_0$. By the assumption and the construction, $\mathcal{C}$ is a strictly antitone subspace of $\mathcal{H} \times \mathcal{H}$ with $0 \leq \dim \mathcal{C} < \dim \mathcal{H}$ and $\mathcal{D}$ is a monotone subspace of $\mathcal{H} \times \mathcal{H}$ with $0 \leq \dim \mathcal{D} \leq \dim \mathcal{H}$. Let $\mathcal{A}_0$ be a maximal monotone linear subspace which contains $\mathcal{D}$. Then

$$\dim(\mathcal{A}_0 + \mathcal{B}_0) = \dim(\mathcal{A}_0 \oplus \mathcal{C}) = \dim \mathcal{A}_0 + \dim \mathcal{C} = \dim \mathcal{H} + \dim \mathcal{C} < \dim(\mathcal{H} \times \mathcal{H}).$$

Hence we can take a point $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \in \mathcal{H} \times \mathcal{H}$ such that $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \notin (\mathcal{A}_0 + \mathcal{B}_0)$. Now let $\mathcal{A} = \{(\mathbf{x} + \tilde{\mathbf{x}} + \mathbf{u}^B, \mathbf{y} + \tilde{\mathbf{y}} + \mathbf{v}^B) : (\mathbf{x}, \mathbf{y}) \in \mathcal{A}_0\}$. By the construction, $\mathcal{A}$ is a maximal monotone affine subspace of $\mathcal{H} \times \mathcal{H}$. If $(\mathbf{x}', \mathbf{y}') \in \mathcal{A} \cap \mathcal{B}$, we would have that

$$\mathcal{B} \ni (\mathbf{x}', \mathbf{y}') = (\mathbf{x} + \tilde{\mathbf{x}} + \mathbf{u}^B, \mathbf{y} + \tilde{\mathbf{y}} + \mathbf{v}^B) \text{ for some } (\mathbf{x}, \mathbf{y}) \in \mathcal{A}_0;$$

hence

$$(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = -(\mathbf{x}, \mathbf{y}) + (\mathbf{x}' - \mathbf{u}^B, \mathbf{y}' - \mathbf{v}^B) \in \mathcal{A}_0 + \mathcal{B}_0.$$

This contradicts $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \notin (\mathcal{A}_0 + \mathcal{B}_0)$. Thus we have shown that $\mathcal{A} \cap \mathcal{B} = \emptyset$. $\square$

3. Main results. We first introduce notation and symbols which we use throughout this section. Let $\mathcal{M}$ denote the n^2 -dimensional linear space of $n \times n$ real matrices, $\mathcal{S}$ the $n(n+1)/2$ -dimensional linear space of $n \times n$ real symmetric matrices, and $\tilde{\mathcal{S}}$ the $n(n-1)/2$ -dimensional linear space of $n \times n$ real skew-symmetric matrices. We use the notation $\text{Tr } \mathbf{A}$ for the trace of a matrix $\mathbf{A} \in \mathcal{M}$ and the notation $\mathbf{A} \bullet \mathbf{B} = \text{Tr } \mathbf{A}^T \mathbf{B}$ for the inner product of $\mathbf{A} \in \mathcal{M}$ and $\mathbf{B} \in \mathcal{M}$. We also use the notation

$$\text{vec } \mathbf{A} = \begin{pmatrix} \mathbf{A}_{\cdot 1} \\ \mathbf{A}_{\cdot 2} \\ \vdots \\ \mathbf{A}_{\cdot n} \end{pmatrix} \in R^{n^2},$$

where $\mathbf{A}_{\cdot j} \in R^n$ denotes the j th column of an $n \times n$ matrix $\mathbf{A}$. Then we see that

$$\mathbf{A} \bullet \mathbf{B} = (\text{vec } \mathbf{A}) \cdot (\text{vec } \mathbf{B}) \text{ for every } \mathbf{A}, \mathbf{B} \in \mathcal{M}.$$

Therefore we may identify $\mathcal{M}$ as the n^2 -dimensional Euclidean space R^{n^2} with the standard inner product, $\mathcal{S}$ as an $n(n+1)/2$ -dimensional linear subspace of R^{n^2} , and $\tilde{\mathcal{S}}$ as an $n(n-1)/2$ -dimensional linear subspace of R^{n^2} . We also see that $\tilde{\mathcal{S}}$ is the orthogonal complement of $\mathcal{S} \subset \mathcal{M}$;

$$(2) \quad \mathcal{S} + \tilde{\mathcal{S}} = \mathcal{M} \text{ and } \mathbf{A} \bullet \mathbf{B} = 0 \text{ for every } \mathbf{A} \in \mathcal{S} \text{ and } \mathbf{B} \in \tilde{\mathcal{S}}.$$

Let $\mathcal{S}_+$ denote the set of positive semidefinite matrices in $\mathcal{S}$ and $\mathcal{S}_{++}$ the set of positive definite matrices in $\mathcal{S}$. Given an $n(n+1)/2$ -dimensional affine subspace $\mathcal{F}$ of $\mathcal{S} \times \mathcal{S}$ satisfying the monotonicity

$$(3) \quad (\mathbf{X}' - \mathbf{X}) \bullet (\mathbf{Y}' - \mathbf{Y}) \geq 0 \text{ for every } (\mathbf{X}', \mathbf{Y}') \in \mathcal{F} \text{ and } (\mathbf{X}, \mathbf{Y}) \in \mathcal{F},$$

we consider the SDLCP:

$$(4) \quad \text{Find an } (\mathbf{X}, \mathbf{Y}) \in \mathcal{F} \text{ such that } \mathbf{X} \in \mathcal{S}_+, \mathbf{Y} \in \mathcal{S}_+, \text{ and } \mathbf{X} \bullet \mathbf{Y} = 0.$$

The monotone SDLCP was introduced by Kojima, Shindoh, and Hara [12] as an extension of the monotone linear complementarity problem (LCP) and a mathematical model for the primal-dual pair of SDPs. They proposed a family of interior-point algorithms for solving the monotone SDLCP (4). Also many primal-dual interior-point algorithms [2, 14, 16, etc.] developed for the SDP can be easily extended to the monotone SDLCP (4). See [12] for a more detailed discussion of the relations between the primal-dual pair of SDPs and the monotone SDLCP.

We investigate five kinds of search directions used in interior-point algorithms for the SDP and the SDLCP. We describe them in terms of the monotone SDLCP (4) and characterize each of them as the intersection of a maximal monotone affine subspace of $\mathcal{S} \times \mathcal{S}$ and a strictly and maximal antitone affine subspace of $\mathcal{S} \times \mathcal{S}$.

Suppose that $(\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_{++} \times \mathcal{S}_{++}$ is the current iterate of an interior-point algorithm for the monotone SDLCP (4). Let $\mu \geq 0$. All the search directions share an affine subspace

$$(5) \quad \mathcal{G} = \{(d\mathbf{X}, d\mathbf{Y}) \in \mathcal{S} \times \mathcal{S} : (\mathbf{X} + d\mathbf{X}, \mathbf{Y} + d\mathbf{Y}) \in \mathcal{F}\}.$$

Let $\mathbf{S} = \mathbf{X}^{1/2}(\mathbf{X}^{1/2}\mathbf{Y}\mathbf{X}^{1/2})^{-1/2}\mathbf{X}^{1/2}$, and let $\tilde{\mathcal{F}}$ be a maximal monotone linear subspace of $\tilde{\mathcal{S}} \times \tilde{\mathcal{S}}$. Define

$$(6) \quad \left. \begin{aligned} \mathcal{L}_1 &= \{(d\mathbf{X}, d\mathbf{Y}) \in \mathcal{S} \times \mathcal{S} : \mathbf{S}^{-1}d\mathbf{X}\mathbf{S}^{-1} + d\mathbf{Y} = \mu\mathbf{X}^{-1} - \mathbf{Y}\}, \\ \mathcal{L}_2 &= \left\{ (d\mathbf{X}, d\mathbf{Y}) \in \mathcal{S} \times \mathcal{S} : \begin{aligned} &(\mathbf{X}^{1/2})^{-1}(\mathbf{X}d\mathbf{Y} + d\mathbf{X}\mathbf{Y})\mathbf{X}^{1/2} \\ &+ \mathbf{X}^{1/2}(d\mathbf{Y}\mathbf{X} + \mathbf{Y}d\mathbf{X})(\mathbf{X}^{1/2})^{-1} \\ &= 2(\mu\mathbf{I} - \mathbf{X}^{1/2}\mathbf{Y}\mathbf{X}^{1/2}) \end{aligned} \right\}, \\ \mathcal{L}_3 &= \{(d\mathbf{X}, d\mathbf{Y}) \in \mathcal{S} \times \mathcal{S} : d\mathbf{Y} + \mu\mathbf{X}^{-1}d\mathbf{X}\mathbf{X}^{-1} = \mu\mathbf{X}^{-1} - \mathbf{Y}\}, \\ \mathcal{L}_4 &= \left\{ (d\mathbf{X}, d\mathbf{Y}) \in \mathcal{S} \times \mathcal{S} : \begin{aligned} &\mathbf{X}(d\mathbf{Y} + \widetilde{d\mathbf{Y}}) + (d\mathbf{X} + \widetilde{d\mathbf{X}})\mathbf{Y} \\ &= \mu\mathbf{I} - \mathbf{X}\mathbf{Y} \\ &\text{for some } (\widetilde{d\mathbf{X}}, \widetilde{d\mathbf{Y}}) \in \tilde{\mathcal{F}} \end{aligned} \right\}, \\ \mathcal{L}_5 &= \left\{ (d\mathbf{X}, d\mathbf{Y}) \in \mathcal{S} \times \mathcal{S} : \begin{aligned} &\mathbf{X}d\mathbf{Y} + d\mathbf{Y}\mathbf{X} + d\mathbf{X}\mathbf{Y} + \mathbf{Y}d\mathbf{X} \\ &= 2\mu\mathbf{I} - (\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X}) \end{aligned} \right\}. \end{aligned} \right\}$$

Then a vector $(d\mathbf{X}, d\mathbf{Y})$ in the intersection of $\mathcal{G}$ with each $\mathcal{L}_i$ ($i = 1, 2, 3, 4, 5$) corresponds to the search direction proposed and studied in the papers [16, 14, 2, 12, and 2], respectively.

The theorem below ensures the existence and the uniqueness of each of these search directions; i.e., $\mathcal{G} \cap \mathcal{L}_i$ is a singleton ($i = 1, 2, 3, 4, 5$). But we should mention that for $i = 1, 2, 3, 4$ the existence and the uniqueness of the search direction corresponding to $\mathcal{G} \cap \mathcal{L}_i$ is already known. See the papers [16, 14, 2, 6, and 12], respectively.

Recall that $\tilde{\mathcal{F}}$ in the definition of $\mathcal{L}_4$ is an arbitrary maximal monotone linear subspace of $\tilde{\mathcal{S}} \times \tilde{\mathcal{S}}$. Hence $\mathcal{G} \cap \mathcal{L}_4$ defines a family of search directions [12]. It is known that the search direction given in the paper [14], which we have described as the unique element in $\mathcal{G} \cap \mathcal{L}_{2,2}$, belongs to this family. More precisely, $\mathcal{G} \cap \mathcal{L}_2 = \mathcal{G} \cap \mathcal{L}_4$ holds if we take $\tilde{\mathcal{F}} = \{\mathbf{O}\} \times \tilde{\mathcal{S}}$. In this case, we can characterize the search direction $(d\mathbf{X}, d\mathbf{Y}) \in \mathcal{G} \cap \mathcal{L}_4$ as the unique solution of

$$\mathbf{X}(d\mathbf{Y} + \widetilde{d\mathbf{Y}}) + d\mathbf{X}\mathbf{Y} = \mu\mathbf{I} - \mathbf{X}\mathbf{Y} \quad \text{and} \quad (d\mathbf{X}, d\mathbf{Y}) \in \mathcal{G} \quad \text{for some } \widetilde{d\mathbf{Y}} \in \tilde{\mathcal{S}}.$$

This search direction was proposed independently by two group of researchers, Helmberg et al. [6] and Kojima, Shindoh, and Hara [12].

The search direction which we have described as the unique element of $\mathcal{G} \cap \mathcal{L}_5$ was proposed by Alizadeh, Haeberly, and Overton [2]. We call it the AHO search direction here. They presented an interior-point method using the AHO search direction for the SDP and reported through numerical results that their method worked very efficiently. However, neither the existence of the AHO search direction nor theoretical convergence of their interior-point method with the use of the AHO direction was discussed in [2]. We will see in the theorem below that the AHO search direction uniquely exists whenever the current iterate $(\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_{++} \times \mathcal{S}_{++}$ satisfies $(\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X}) \in \mathcal{S}_+$. [11] studies the global and local convergence of a predictor-corrector interior-point method using the AHO search direction for the monotone SDLCP.

THEOREM 3.1. *Let $\mathcal{F}$ be a maximal monotone affine subspace of $\mathcal{S} \times \mathcal{S}$ and $\mathcal{G}$ be the affine subspace defined in (5). Assume that $(\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_{++} \times \mathcal{S}_{++}$ and $\mu \geq 0$. Define the affine subspaces $\mathcal{L}_i$ ($i = 1, 2, \dots, 5$) as in (6).*

- (i) $\mathcal{G}$ is a maximal monotone affine subspace of $\mathcal{S} \times \mathcal{S}$.
- (ii) For $i = 1, 2, 3, 4$, $\mathcal{L}_i$ is a strictly and maximal antitone affine subspace of $\mathcal{S} \times \mathcal{S}$ and $\mathcal{G} \cap \mathcal{L}_i$ is a singleton.
- (iii) If $(\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X}) \in \mathcal{S}_+$, then $\mathcal{L}_5$ is a strictly and maximal antitone affine subspace of $\mathcal{S} \times \mathcal{S}$ and $\mathcal{G} \cap \mathcal{L}_5$ is a singleton.

Remark. The requirement $(\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X}) \in \mathcal{S}_+$ in (iii) of Theorem 3.1 holds if $(\mathbf{X}, \mathbf{Y})$ lies in some neighborhood of the central manifold

$$\{(\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_{++} \times \mathcal{S}_{++} : \mathbf{X}\mathbf{Y} = \mu \mathbf{I} \text{ for some } \mu > 0\}$$

such as

$$\left\{ (\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_{++} \times \mathcal{S}_{++} : \left\| \frac{1}{2}(\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X}) - \mu \mathbf{I} \right\|_F \leq \gamma \mu \text{ for some } \mu > 0 \right\} \text{ or } \left\{ (\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_{++} \times \mathcal{S}_{++} : \mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X} \succeq (1 - \gamma)\mu \mathbf{I} \text{ for some } \mu > 0 \right\},$$

where $\gamma \in [0, 1)$. See also [11]. $\square$

Let $\mathbf{A} \otimes \mathbf{B}$ denote the Kronecker product of matrices $\mathbf{A}$ and $\mathbf{B}$. In the proof of Theorem 3.1, we will utilize the following lemma repeatedly. We also use the relation $(\mathbf{A} \otimes \mathbf{B})\text{vec } \mathbf{M} = \text{vec } (\mathbf{B}\mathbf{M}\mathbf{A}^T)$. See, e.g., [4] for its proof and some other basic properties on the Kronecker product of matrices.

LEMMA 3.2. *Let λ_i ($i = 1, 2, \dots, n$) and μ_j ($j = 1, 2, \dots, n$) be the eigenvalues of $\mathbf{A} \in \mathcal{M}$ and $\mathbf{B} \in \mathcal{M}$, respectively. Then $\lambda_i \mu_j$ ($i = 1, 2, \dots, n$, $j = 1, 2, \dots, n$) are the eigenvalues of $\mathbf{A} \otimes \mathbf{B}$. If in addition $\mathbf{A} \in \mathcal{S}_{++}$ and $\mathbf{B} \in \mathcal{S}_{++}$, then $\mathbf{A} \otimes \mathbf{B}$ is symmetric and positive definite.*

Proof of Theorem 3.1. The assertion (i) follows directly from the maximal monotonicity of $\mathcal{F}$. To prove the assertions (ii) and (iii), we rewrite the affine subspaces $\mathcal{L}_i$ ($i = 1, 2, 3, 4, 5$) using the Kronecker product as follows:

$$\begin{aligned} \mathcal{L}_i &= \{ (d\mathbf{X}, d\mathbf{Y}) \in \mathcal{S} \times \mathcal{S} : \text{vec } d\mathbf{Y} = \mathbf{Q}_i \text{vec } d\mathbf{X} + \mathbf{q}_i \} \quad (i = 1, 2, 3, 5), \\ \mathcal{L}_4 &= \left\{ (d\mathbf{X}, d\mathbf{Y}) \in \mathcal{S} \times \mathcal{S} : \begin{array}{l} \text{vec } (d\mathbf{Y} + \widetilde{d\mathbf{Y}}) = \mathbf{Q}_4 \text{vec } (d\mathbf{X} + \widetilde{d\mathbf{X}}) + \mathbf{q}_4 \\ \text{for some } (\widetilde{d\mathbf{X}}, \widetilde{d\mathbf{Y}}) \in \widetilde{\mathcal{F}} \end{array} \right\}, \end{aligned} \quad (7)$$

where

$$(8) \quad \left. \begin{aligned} \mathbf{Q}_1 &= -\mathbf{S}^{-1} \otimes \mathbf{S}^{-1}, \quad \mathbf{q}_1 = \text{vec} (\mu \mathbf{X}^{-1} - \mathbf{Y}), \\ \mathbf{Q}_2 &= -\frac{\mathbf{Y} \otimes \mathbf{X}^{-1} + \mathbf{X}^{-1} \otimes \mathbf{Y}}{2}, \quad \mathbf{q}_2 = \text{vec} (\mu \mathbf{X}^{-1} - \mathbf{Y}), \\ \mathbf{Q}_3 &= -\mu \mathbf{X}^{-1} \otimes \mathbf{X}^{-1}, \quad \mathbf{q}_3 = \text{vec} (\mu \mathbf{X}^{-1} - \mathbf{Y}), \\ \mathbf{Q}_4 &= -\mathbf{Y} \otimes \mathbf{X}^{-1}, \quad \mathbf{q}_4 = \text{vec} (\mu \mathbf{X}^{-1} - \mathbf{Y}), \\ \mathbf{Q}_5 &= -(\mathbf{I} \otimes \mathbf{X} + \mathbf{X} \otimes \mathbf{I})^{-1}(\mathbf{I} \otimes \mathbf{Y} + \mathbf{Y} \otimes \mathbf{I}), \\ \mathbf{q}_5 &= \text{vec} (2\mu \mathbf{I} - (\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X})). \end{aligned} \right\}$$

(Note that the nonsingularity of the matrix $(\mathbf{I} \otimes \mathbf{X} + \mathbf{X} \otimes \mathbf{I})$ follows from $\mathbf{X} \in \mathcal{S}_{++}$ and Lemma 3.2). We know from the definition of $\mathcal{L}_i$ ($i = 1, 2, 3$) in (6) that

$$d\mathbf{Y} \in \mathcal{S} \text{ if } \text{vec } d\mathbf{Y} = \mathbf{Q}_i(\text{vec } d\mathbf{X}) + \mathbf{q}_i \text{ and } d\mathbf{X} \in \mathcal{S} \text{ (} i = 1, 2, 3 \text{);}$$

hence we may regard $\mathbf{Q}_i$ as a linear operator on $\mathcal{S}$. We also see from $\mathbf{X} \in \mathcal{S}_{++}$, $\mathbf{Y} \in \mathcal{S}_{++}$, $\mathbf{S} \in \mathcal{S}_{++}$, and Lemma 3.2 that $\mathbf{Q}_i$ is negative definite on $\{\text{vec } \mathbf{U} : \mathbf{U} \in \mathcal{S}\}$ ($i = 1, 2, 3$). Thus we obtain by Corollary 2.2 that $\mathcal{L}_i$ ($i = 1, 2, 3$) are strictly and maximal antitone subspaces of $\mathcal{S} \times \mathcal{S}$.

Now we consider the case of $\mathcal{L}_4$. Define

$$(9) \quad \widehat{\mathcal{L}}_4(\mathbf{q}) = \{(\widehat{d\mathbf{X}}, \widehat{d\mathbf{Y}}) \in \mathcal{M} \times \mathcal{M} : \text{vec } \widehat{d\mathbf{Y}} = \mathbf{Q}_4 \text{vec } \widehat{d\mathbf{X}} + \mathbf{q}\} \text{ for every } \mathbf{q} \in \mathbf{R}^{n \times n}.$$

Since $\mathbf{Q}_4$ is negative definite by Lemma 3.2, we know by Corollary 2.2 that $\widehat{\mathcal{L}}_4(\mathbf{q})$ is a strictly and maximal antitone affine subspace of $\mathcal{M} \times \mathcal{M}$ for every $\mathbf{q} \in \mathbf{R}^{n \times n}$. Using this fact we will show that $\mathcal{L}_4$ is a strictly antitone affine subspace of $\mathcal{S} \times \mathcal{S}$. Suppose that $(d\mathbf{X}_1, d\mathbf{Y}_1) \in \mathcal{L}_4$, $(d\mathbf{X}_2, d\mathbf{Y}_2) \in \mathcal{L}_4$, and $(d\mathbf{X}_1, d\mathbf{Y}_1) \neq (d\mathbf{X}_2, d\mathbf{Y}_2)$. Then there exists an $(\widetilde{d\mathbf{X}}, \widetilde{d\mathbf{Y}}) \in \widetilde{\mathcal{F}}$ such that $(d\mathbf{X}_1 - d\mathbf{X}_2 + \widetilde{d\mathbf{X}}, d\mathbf{Y}_1 - d\mathbf{Y}_2 + \widetilde{d\mathbf{Y}}) \in \widehat{\mathcal{L}}_4(\mathbf{0})$. Since $(d\mathbf{X}_1 - d\mathbf{X}_2 + \widetilde{d\mathbf{X}}, d\mathbf{Y}_1 - d\mathbf{Y}_2 + \widetilde{d\mathbf{Y}}) \neq \mathbf{0}$ and $\widehat{\mathcal{L}}_4(\mathbf{0})$ is a strictly and maximal antitone linear subspace of $\mathcal{M} \times \mathcal{M}$, we have that

$$\begin{aligned} 0 &> (d\mathbf{X}_1 - d\mathbf{X}_2 + \widetilde{d\mathbf{X}}) \bullet (d\mathbf{Y}_1 - d\mathbf{Y}_2 + \widetilde{d\mathbf{Y}}) \\ &= (d\mathbf{X}_1 - d\mathbf{X}_2) \bullet (d\mathbf{Y}_1 - d\mathbf{Y}_2) + \widetilde{d\mathbf{X}} \bullet \widetilde{d\mathbf{Y}} \\ &\quad (\text{since } (d\mathbf{X}_1 - d\mathbf{X}_2), (d\mathbf{Y}_1 - d\mathbf{Y}_2) \in \mathcal{S} \text{ and } \widetilde{d\mathbf{X}}, \widetilde{d\mathbf{Y}} \in \widetilde{\mathcal{S}} \text{ (see (2))}) \\ &\geq (d\mathbf{X}_1 - d\mathbf{X}_2) \bullet (d\mathbf{Y}_1 - d\mathbf{Y}_2) \text{ (since } \widetilde{\mathcal{F}} \text{ is monotone).} \end{aligned}$$

Thus we have shown that $\mathcal{L}_4$ is a strictly antitone affine subspace of $\mathcal{S} \times \mathcal{S}$. To derive its maximality in $\mathcal{S} \times \mathcal{S}$, we show that for every $\overline{d\mathbf{X}} \in \mathcal{S}$ there exists a unique $\overline{d\mathbf{Y}} \in \mathcal{S}$ for which $(\overline{d\mathbf{X}}, \overline{d\mathbf{Y}}) \in \mathcal{L}_4$; this implies $\dim \mathcal{L}_4 = \dim \mathcal{S}$, hence the maximality in $\mathcal{S} \times \mathcal{S}$ follows from Lemma 2.1. Let $\overline{d\mathbf{X}} \in \mathcal{S}$ be fixed. Consider

$$\mathcal{K} = \{(\overline{d\mathbf{X}} + \widetilde{d\mathbf{X}}, d\mathbf{Y} + \widetilde{d\mathbf{X}}) \in \mathcal{M} \times \mathcal{M} : d\mathbf{Y} \in \mathcal{S}, (\widetilde{d\mathbf{X}}, \widetilde{d\mathbf{Y}}) \in \widetilde{\mathcal{F}}\}.$$

Then $\mathcal{K}$ forms an n^2 -dimensional monotone affine subspace of $\mathcal{M} \times \mathcal{M}$. Applying Lemma 2.3, we obtain that there exists a unique $(\overline{d\mathbf{X}} + \widetilde{d\mathbf{X}}, \overline{d\mathbf{Y}} + \widetilde{d\mathbf{X}}) \in \mathcal{K} \cap \widehat{\mathcal{L}}_4(\mathbf{q}_4)$, which implies that $(\overline{d\mathbf{X}}, \overline{d\mathbf{Y}}) \in \mathcal{L}_4$.

We now deal with the case of $\mathcal{L}_5$. If a pair of $d\mathbf{X} \in \mathcal{S}$ and $d\mathbf{Y} \in \mathcal{M}$ satisfies

$$\mathbf{X}d\mathbf{Y} + d\mathbf{Y}\mathbf{X} + d\mathbf{X}\mathbf{Y} + \mathbf{Y}d\mathbf{X} = 2\mu\mathbf{I} - (\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X}),$$

then the pair of $d\mathbf{X} \in \mathcal{S}$ and $d\mathbf{Y}^T \in \mathcal{M}$ does. But we can rewrite this identity as $\text{vec } d\mathbf{Y} = \mathbf{Q}_5 \text{vec } d\mathbf{X} + \mathbf{q}_5$. Hence $d\mathbf{Y} = d\mathbf{Y}^T \in \mathcal{M}$ if $\text{vec } d\mathbf{Y} = \mathbf{Q}_5 \text{vec } d\mathbf{X} + \mathbf{q}_5$ and $d\mathbf{X} \in \mathcal{S}$. This implies that $\text{vec } d\mathbf{Y} = \mathbf{Q}_5 \text{vec } d\mathbf{X} + \mathbf{q}_5 \in \{\text{vec } \mathbf{U} : \mathbf{U} \in \mathcal{S}\}$ for every $d\mathbf{X} \in \mathcal{S}$ and that $\dim \mathcal{L}_5 = \dim \mathcal{S}$. We have left to show that $\mathbf{Q}_5$ is negative definite on $\{\text{vec } \mathbf{U} : \mathbf{U} \in \mathcal{S}\}$. Let $\mathbf{O} \neq \mathbf{U} \in \mathcal{S}$ and $\mathbf{W}$ be the nonzero matrix in $\mathcal{S}$ uniquely determined by $\mathbf{X}\mathbf{W} + \mathbf{W}\mathbf{X} = \mathbf{U}$; i.e., $\text{vec } \mathbf{W} = (\mathbf{I} \otimes \mathbf{X} + \mathbf{X} \otimes \mathbf{I})^{-1} \text{vec } \mathbf{U}$. Then

$$\begin{aligned} & \text{vec } \mathbf{U} \cdot (\mathbf{Q}_5 \text{vec } \mathbf{U}) \\ &= -\text{vec } \mathbf{U} \cdot ((\mathbf{I} \otimes \mathbf{X} + \mathbf{X} \otimes \mathbf{I})^{-1} (\mathbf{I} \otimes \mathbf{Y} + \mathbf{Y} \otimes \mathbf{I}) \text{vec } \mathbf{U}) \\ &= -\text{vec } \mathbf{W} \cdot ((\mathbf{I} \otimes \mathbf{Y} + \mathbf{Y} \otimes \mathbf{I})(\mathbf{I} \otimes \mathbf{X} + \mathbf{X} \otimes \mathbf{I}) \text{vec } \mathbf{W}) \\ &= -\text{vec } \mathbf{W} \cdot ((\mathbf{X} \otimes \mathbf{Y} + \mathbf{Y} \otimes \mathbf{X} + \mathbf{Y}\mathbf{X} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{Y}\mathbf{X}) \text{vec } \mathbf{W}) \\ &< -\text{vec } \mathbf{W} \cdot ((\mathbf{Y}\mathbf{X} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{Y}\mathbf{X}) \text{vec } \mathbf{W}) \\ &\quad (\text{since } \mathbf{X} \otimes \mathbf{Y} \text{ and } \mathbf{Y} \otimes \mathbf{X} \text{ are positive definite by Lemma 3.2}) \\ &= -\mathbf{W} \bullet (\mathbf{W}\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X}\mathbf{W}) \\ &= -\text{Tr } \mathbf{W}(\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X})\mathbf{W} \\ &\leq 0. \end{aligned}$$

Here the last inequality follows from the assumption that $\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X} \in \mathcal{S}_+$.

Finally we obtain by Lemma 2.3 that $\mathcal{G} \cap \mathcal{L}_i$ is a singleton ($i = 1, 2, 3, 4, 5$). This completes the proof of Theorem 3.1. $\square$

4. Concluding remarks on Theorem 3.1.

(A) For each $\mu > 0$, define

$$\mathcal{C}(\mu) = \{(\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_+ \times \mathcal{S}_+ : \mathbf{X}\mathbf{Y} = \mu\mathbf{I}\}.$$

We can prove that $\mathcal{C}(\mu)$ is a strictly and maximal antitone smooth manifold in $\mathcal{S} \times \mathcal{S}$ for every μ . Each point of the central trajectory of the monotone SDLCP (4) is uniquely determined by the intersection of the maximal monotone affine subspace $\mathcal{F}$ and the strictly and maximal antitone smooth manifold $\mathcal{C}(\mu)$. Our characterization of five types of search directions used in the SDP and the monotone SDLCP (4) is founded on this fact. The systems of linear equations involved in $\mathcal{L}_i$ ($i = 2, 3, 4, 5$) (see (6)) were derived as distinct (scaled) linear approximations of the system of equations $\mathbf{X}\mathbf{Y} = \mu\mathbf{I}$ at $(\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_{++} \times \mathcal{S}_{++}$ [1, 12, 14], while the system of linear equations involved in $\mathcal{L}_1$ (see (6)) was derived from the self-concordant barrier function [15, 16]. When $(\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_{++} \times \mathcal{S}_{++}$ lies on $\mathcal{C}(\mu)$, all $\mathcal{L}_i$ ($i = 1, 2, 3, 4, 5$) coincide with the tangent space of $\mathcal{C}(\mu)$ at $(\mathbf{X}, \mathbf{Y}) \in \mathcal{C}(\mu)$,

$$\begin{aligned} & \{(d\mathbf{X}, d\mathbf{Y}) \in \mathcal{S} \times \mathcal{S} : \mathbf{X}d\mathbf{Y} + d\mathbf{X}\mathbf{Y} = \mathbf{O}\} \\ &= \{(d\mathbf{X}, d\mathbf{Y}) \in \mathcal{S} \times \mathcal{S} : \mu \text{vec } d\mathbf{Y} = -(\mathbf{Y} \otimes \mathbf{Y}) \text{vec } d\mathbf{X}\}, \end{aligned}$$

which inherits the strict and maximal antitonicity from $\mathcal{C}(\mu)$. It is shown in the recent paper [17] that each point on the central trajectory of the monotone generalized complementarity problem [7] is characterized as the intersection of a maximal monotone subset associated with the feasible region and a strictly and maximal antitone subset, as in the case of the SDLCP. Applying the basic lemmas given in section 2, we can

derive the similar characterization on some other search directions used in interior-point methods [9, 10, 16, 20, etc.] for various problems covered by the monotone generalized complementarity problem.

(B) The strict and maximal antitonicity of the affine subspace $\mathcal{L}_i$ of $\mathcal{S} \times \mathcal{S}$ is necessary to ensure $\mathcal{G} \cap \mathcal{L}_i \neq \emptyset$ for any maximal monotone affine subspace $\mathcal{F} \subset \mathcal{S} \times \mathcal{S}$ ($i = 1, 2, 3, 4, 5$). In other words, if the affine subspace $\mathcal{L}_i$ does not contain a strictly and maximal antitone affine subspace, then by (ii) of Lemma 2.3, there would be a maximal monotone affine subspace $\mathcal{F}$ and an $(\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_{++} \times \mathcal{S}_{++}$ such that $\mathcal{G} \cap \mathcal{L}_i = \emptyset$, where $\mathcal{G} = \{(d\mathbf{X}, d\mathbf{Y}) \in \mathcal{S} \times \mathcal{S} : (\mathbf{X} + d\mathbf{X}, \mathbf{Y} + d\mathbf{Y}) \in \mathcal{F}\}$.

(C) The assumption $(\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_{++} \times \mathcal{S}_{++}$ does not imply $(\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X}) \in \mathcal{S}_{++}$ in (iii) of Theorem 3.1. To see this, consider the pair of matrices

$$\mathbf{X} = \begin{pmatrix} 49 & 0 \\ 0 & 1 \end{pmatrix} \in \mathcal{S}_{++} \quad \text{and} \quad \mathbf{Y} = \begin{pmatrix} 4 & 1 \\ 1 & 1 \end{pmatrix} \in \mathcal{S}_{++}.$$

In this example, we have that

$$\mathbf{X}\mathbf{Y} + \mathbf{Y}\mathbf{X} = \begin{pmatrix} 392 & 50 \\ 50 & 2 \end{pmatrix} \notin \mathcal{S}_{+}.$$

Furthermore the matrix

$$\mathbf{Q}_5 = -(I \otimes \mathbf{X} + \mathbf{X} \otimes I)^{-1}(I \otimes \mathbf{Y} + \mathbf{Y} \otimes I) = - \begin{pmatrix} \frac{4}{49} & \frac{1}{98} & \frac{1}{98} & 0 \\ \frac{1}{50} & \frac{1}{10} & 0 & \frac{1}{50} \\ \frac{1}{50} & 0 & \frac{1}{10} & \frac{1}{50} \\ 0 & \frac{1}{2} & \frac{1}{2} & 1 \end{pmatrix}$$

is not negative definite on $\{\mathbf{vec} \, \mathbf{U} : \mathbf{U} \in \mathcal{S}\}$ since we have

$$(\mathbf{vec} \, \mathbf{W})^T \mathbf{Q}_5 \mathbf{vec} \, \mathbf{W} = \frac{7}{5} > 0 \quad \text{for} \quad \mathbf{W} = \begin{pmatrix} 0 & -5 \\ -5 & 2 \end{pmatrix} \in \mathcal{S}.$$

This implies that $\mathcal{L}_5$ is not antitone; then $\mathcal{L}_5$ does not contain maximal antitone affine subspace. In view of remark (B) above, we can take a maximal monotone affine subspace $\mathcal{F} \in \mathcal{S} \times \mathcal{S}$ and an $(\mathbf{X}, \mathbf{Y}) \in \mathcal{S}_{++} \times \mathcal{S}_{++}$ such that $\mathcal{L}_5 \cap \mathcal{G} = \emptyset$. See also [19].

(D) The proof of the assertion that $\mathcal{L}_4$ is a maximal antitone affine subspace of $\mathcal{S} \times \mathcal{S}$ is a little bit complicated. If we are only concerned with the assertion that $\mathcal{G} \cap \mathcal{L}_4$ is a singleton, we can give a simple proof. Define $\widehat{\mathcal{L}}_4(\mathbf{q})$ by (9) and $\widehat{\mathcal{G}}$ by

$$\widehat{\mathcal{G}} = \{(d\mathbf{X} + \widetilde{d\mathbf{X}}, d\mathbf{Y} + \widetilde{d\mathbf{Y}}) : (d\mathbf{X}, d\mathbf{Y}) \in \mathcal{G}, (\widetilde{d\mathbf{X}}, \widetilde{d\mathbf{Y}}) \in \widetilde{\mathcal{F}}\},$$

then $\widehat{\mathcal{L}}_4(\mathbf{q}_4)$ is a strictly and maximal antitone affine subspace of $\mathcal{M} \times \mathcal{M}$, and $\widehat{\mathcal{G}}$ is a maximal monotone affine subspace of $\mathcal{M} \times \mathcal{M}$. By Lemma 2.3, there exists a unique $(\widetilde{d\mathbf{X}}^*, \widetilde{d\mathbf{Y}}^*) \in \widehat{\mathcal{G}} \cap \widehat{\mathcal{L}}_4(\mathbf{q}_4)$. On the other hand, we see by construction that $(d\mathbf{X}, d\mathbf{Y}) \in \mathcal{G} \cap \mathcal{L}_4$ if and only if $(d\mathbf{X} + \widetilde{d\mathbf{X}}, d\mathbf{Y} + \widetilde{d\mathbf{Y}}) \in \widehat{\mathcal{L}}_4(\mathbf{q}_4)$ for some $(\widetilde{d\mathbf{X}}, \widetilde{d\mathbf{Y}}) \in \widetilde{\mathcal{F}}$. Therefore $\mathcal{G} \cap \mathcal{L}_4$ is a singleton.

Acknowledgment. The authors are grateful to anonymous referees for their valuable comments.

REFERENCES

- [1] F. ALIZADEH, *Interior point methods in semidefinite programming with applications to combinatorial optimization*, SIAM J. Optim., 5 (1995), pp. 13–51.
- [2] F. ALIZADEH, J.-P. HAEGERLY, AND M. OVERTON, *Primal-Dual Interior-Point Methods for Semidefinite Programming*, Tech. report 659, Computer Science Department, Courant Institute of Mathematical Sciences, New York University, New York, 1994.
- [3] J. AUBIN AND I. EKELAND, *Applied Nonlinear Analysis*, John Wiley and Sons, New York, 1984.
- [4] A. GRAHAM, *Kronecker Products and Matrix Calculus with Applications*, Ellis Horwood Limited, John Wiley & Sons, New York, 1981.
- [5] O. GÜLER, *Generalized linear complementarity problems*, Math. Oper. Res., 20 (1995), pp. 441–448.
- [6] C. HELMBERG, F. RENDL, R. VANDERBEI, AND H. WOLKOWICZ, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.
- [7] S. KARAMARDIAN, *Complementarity problems over cones with monotone and pseudomonotone maps*, J. Optim. Theory Appl., 18 (1976), pp. 445–454.
- [8] M. KOJIMA, N. MEGIDDO, T. NOMA, AND A. YOSHISE, *A Unified Approach to Interior Point Algorithms for Linear Complementarity Problems*, Springer-Verlag, New York, 1991.
- [9] M. KOJIMA, S. MIZUNO, AND A. YOSHISE, *A polynomial-time algorithm for a class of linear complementarity problems*, Math. Programming, 44 (1989), pp. 1–26.
- [10] M. KOJIMA, S. MIZUNO, AND A. YOSHISE, *A primal-dual interior point algorithm for linear programming*, in Progress in Mathematical Programming, Interior-Point and Related Methods, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 29–47.
- [11] M. KOJIMA, M. SHIDA, AND S. SHINDOH, *A predictor-corrector interior-point algorithm for the semidefinite linear complementarity problem using the Alizadeh-Haeberly-Overton search direction*, SIAM J. Optim., to appear.
- [12] M. KOJIMA, S. SHINDOH, AND S. HARA, *Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.
- [13] G. MINTY, *Monotone (nonlinear) operators in Hilbert space*, Duke Math. J., 29 (1962), pp. 341–346.
- [14] R. MONTEIRO, *Primal-dual path-following algorithms for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 663–678.
- [15] Y. NESTEROV AND A. NEMIROVSKII, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, PA, 1994.
- [16] Y. NESTEROV AND M. TODD, *Primal-dual interior-point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.
- [17] M. SHIDA, S. SHINDOH, AND M. KOJIMA, *Centers of monotone generalized complementarity problems*, Math. Oper. Res., 22 (1997), pp. 969–976.
- [18] K. TANABE, *Centered Newton methods for mathematical programming*, in System Modeling and Optimization, M. Iri and K. Yajima, eds., Springer-Verlag, New York, 1988, pp. 197–206.
- [19] M. TODD, K. TOH, AND R. TÜTÜNCÜ, *On the Nesterov-Todd direction in semidefinite programming*, SIAM J. Optim., to appear.
- [20] T. TSUCHIYA, *A New Family of Polynomial-Time Interior Point Algorithms for Linear Programming*, Tech. report, The Institute of Statistical Mathematics, Minato-ku, Tokyo, Japan, 1994.
- [21] L. VANDENBERGHE AND S. BOYD, *A primal-dual potential reduction method for problems involving matrix inequalities*, Math. Programming, 69 (1995), pp. 205–236.

INTERIOR POINT ALGORITHMS FOR LINEAR COMPLEMENTARITY PROBLEMS BASED ON LARGE NEIGHBORHOODS OF THE CENTRAL PATH*

GONGYUN ZHAO†

Abstract. In this paper we study a first-order and a high-order algorithm for solving linear complementarity problems. These algorithms are implicitly associated with a large neighborhood whose size may depend on the dimension of the problems. The complexity of these algorithms depends on the size of the neighborhood. For the first-order algorithm, we achieve the complexity bound which the typical large-step algorithms possess. It is well known that the complexity of large-step algorithms is greater than that of short-step ones. By using high-order power series (hence the name high-order algorithm), the iteration complexity can be reduced. We show that the complexity upper bound for our high-order algorithms is equal to that for short-step algorithms.

Key words. interior point algorithm, high-order power series, large neighborhood, large step, complexity, linear complementarity problem

AMS subject classifications. 90C05, 90C60, 49M35

PII. S1052623494275574

1. Introduction. As is well known, all practically effective interior point algorithms use large neighborhoods which allow iterates to move more freely so as to possibly take large steps (thus they are called *large-step algorithms*¹ in the literature). Some works on large-step interior point methods are listed in the references. Among them are [2], [3], [4], [5], [8], [10], [12], [13], [14], [16], [20], and [23]. It is well known that the upper bound of the complexity for large-step algorithms is, in general, greater than that for short-step ones (cf. [1], [15], [18], [19], and [21] for short-step algorithms). Using high-order approximations, the iteration complexity upper bound for large-step algorithms can be reduced and usually this tends to the upper bound for short-step ones as the order tends to infinity. This idea was first presented by Monteiro, Adler, and Resende [13] and then also studied by Zhang and Zhang [20] and Hung and Ye [4].

The contributions of this paper are as follows. We present algorithms based on a new type of large neighborhoods. We show that the complexity of our first-order algorithm is as good as the best complexity bound of existing large-step algorithms; i.e., $O(n|\ln \epsilon|)$. We show how the complexity of our algorithm can be reduced by using high-order derivatives. For instance, if we use the derivatives up to order $\nu = \lceil n^{0.1} \rceil$, then the complexity of our algorithm is bounded by $O(\sqrt{n}|\ln \epsilon|)$.² Note that if, e.g., $n = 10^6$, then $\nu = 4$ is enough. An algorithm of order 4 can be easily implemented.

*Received by the editors October 14, 1994; accepted for publication (in revised form) January 16, 1997.

<http://www.siam.org/journals/siopt/8-2/27557.html>

†Department of Mathematics, National University of Singapore, 119260, Singapore, (matzgy@math.nus.edu.sg).

¹This generally accepted terminology is not accurate because these algorithms do not always take large step lengths. Actually, the step lengths may sometimes be very short. This is the reason that the complexity bounds of large-step algorithms are, in general, worse than that of short-step algorithms.

²See (1.3) for details. The complexity bound (1.3) is essentially due to C. Roos, who, in a private conversation, told the author how to derive (1.3) from the complexity bound (1.2). Independently, the author had found the result (1.3) by using a different proof for the case $\nu = \sqrt{n}$ before C. Roos told him. The author's proof is complicated and cannot be extended to the general case, hence only C. Roos's proof idea is used in the paper.

Let us give a more detailed description about the algorithms and their complexity. Let $z = (x, s) \in R^{2n}$ be an interior point and $\mu > 0$ be a parameter. We denote $u = (xs/\mu)^{1/2}$ (the operation xs is defined in section 2), and define the following functions:

$$\delta(z, \mu) = \|u - u^{-1}\|_2, \quad \zeta(z, \mu) = \|u^{-1}\|_\infty.$$

The function $\delta(z, \mu)$ is called the *distance* from the point z to the point $z^c(\mu)$ on the central trajectory at μ . This definition of distance was first introduced by Jansen et al. in [5] and was also used in [6]. This distance is very suitable for large-neighborhood algorithms, since the distance $\delta(z, \mu) \rightarrow \infty$ if and only if z tends to the boundary of the feasible region. The function $\zeta(z, \mu)$ measures how far the components of u are bounded away from zero.

A typical iteration of our algorithms is as follows. Let $z^{[k]}$ and μ_k be the current feasible point and parameter, respectively, with $\delta(z^{[k]}, \mu_k) = \delta_*$. Here we let $\delta_* = O(n^\tau)$ for some number $\tau \geq 0$. We define a trajectory linking the current point $z^{[k]}$ and the target point $z^c(\mu_k)$. Along the tangent line or a high-order approximation (truncated power series) of the trajectory, we find the next point $z^{[k+1]}$ by minimizing the distance. So we have $\delta(z^{[k+1]}, \mu_k) < \delta(z^{[k]}, \mu_k)$. Then the parameter μ is reduced to μ_{k+1} such that $\delta(z^{[k+1]}, \mu_{k+1}) = \delta_*$ again.

The first order algorithm (Algorithm 3.1) needs at most

$$(1.1) \quad O(n^{\tau+\omega} \ln(\mu_0 n^{2\omega}/\epsilon))$$

iterations starting from a parameter μ_0 to find a feasible point (x, s) with $\langle x, s \rangle \leq \epsilon$, where $\omega = \max\{\tau, 1/2\}$. If $\tau = 1/2$, the complexity bound is $O(n)$ (omitting the $\ln$ -term). This is the best complexity bound to date for large-step algorithms. If $\tau > 1/2$, the complexity bound is $O(n^{2\tau})$, which reflects the dependence of the complexity on the size of the neighborhood.

In the high-order algorithm (Algorithm 3.2) an additional restriction on $z^{[k+1]}$ requiring $\zeta(z^{[k+1]}, \mu_k) \leq \zeta_*$ should be imposed, where ζ_* can be designated to any positive number and may depend on n . The iteration complexity of the algorithm of order ν is bounded by

$$(1.2) \quad O(\zeta_*^{\frac{\nu-1}{\nu}} n^{\frac{\tau}{\nu}+\omega} \ln(\mu_0 n^{2\omega}/\epsilon)).$$

If we let the order $\nu = \lceil n^\gamma \rceil$ for some constant $\gamma > 0$, e.g., $\gamma = 0.1$, then we can derive from (1.2) a complexity bound of

$$(1.3) \quad O(\zeta_*^{\frac{\nu-1}{\nu}} n^\omega \ln(\mu_0 n^{2\omega}/\epsilon))$$

because

$$n^{\tau/\nu} \leq e^{\tau/(\gamma\epsilon)} \quad \forall n \geq 1.$$

(See also the footnote 2 on page 1.) If we consider a special case setting $\zeta_* = O(1)$ and $\tau = 1/2$, then the complexity bound is $O(n^{\frac{1}{2}})$, which is the well-known best upper bound of the iteration complexity for interior point algorithms. We can see that for an appropriate γ (e.g., $\gamma = 0.1$), the order of the algorithm, $\nu = \lceil n^\gamma \rceil$, is fairly small even for large-scale problems. Thus the complexity bound (1.3) explains very well why algorithms using only two to five derivatives can significantly reduce the number of iterations; cf. [7], [11].

The paper is organized as follows. In section 2, we define notations which are used throughout this paper. In section 3, we present the first-order and high-order algorithms. Section 4 collects several lemmas prepared for the complexity analysis. In section 5, the final results on the complexity of the algorithms are presented.

2. Notations. Throughout this paper we use capital letters, English letters, and Greek letters as matrices, column vectors, and scalars, respectively. For any vector x , we denote by x_i the i th component of vector x . We denote $e = (1, \dots, 1)^T$ regardless of dimension.

For any $x, s \in R^n$ and $\alpha \in R$ we define

$$xs = (x_1 s_1, \dots, x_n s_n)^T, \quad x^\alpha = (x_1^\alpha, \dots, x_n^\alpha)^T, \quad |x| = (|x_1|, \dots, |x_n|)^T.$$

$$\langle x, s \rangle = \sum_{i=1}^n x_i s_i, \quad \|x\|_2 = \left(\sum_{i=1}^n x_i^2 \right)^{1/2}, \quad \|x\|_\infty = \max_{1 \leq i \leq n} |x_i|.$$

We specify the following symbols:

$e = (1, \dots, 1)^T$;

x, s : variables in problems (LCP) or in system $(CT)_\mu$ defined in section 3;

$\mu > 0$: the parameter in system $(CT)_\mu$;

$\nu \geq 1$: order of algorithm;

$z = (x, s)$;

$z^c(\mu) = (x^c(\mu), s^c(\mu))$: the μ -center defined by $(CT)_\mu$; the central trajectory consists of all μ -centers for μ ;

$z(\alpha) = (x(\alpha), s(\alpha))$: the trajectory linking a point (x, s) to the point $(x^c(\mu), s^c(\mu))$;

$d_z^{(i)} = (d_x^{(i)}, d_s^{(i)})$: derivatives of $z(\alpha)$ defined by (3.5) and (3.6);

$h_k = \mu^{-1} \sum_{i=k-\nu}^{\nu} d_x^{(i)} d_s^{(k-i)}$;

$u = (xs/\mu)^{1/2}$;

$u(\alpha) = \left(\frac{(x + \sum_{i=1}^{\nu} d_x^{(i)} \alpha^i)(s + \sum_{i=1}^{\nu} d_s^{(i)} \alpha^i)}{\mu} \right)^{1/2}$;

$w(\alpha), r(\alpha)$: defined by (4.4);

$\delta(z, \mu) = \|u - u^{-1}\|_2$: distance from z to $z^c(\mu)$, defined by (3.2);

$\delta = \delta(z, \mu)$;

$\delta(\alpha) = \|u(\alpha) - u(\alpha)^{-1}\|_2$;

$\zeta(z, \mu) = \|u^{-1}\|_\infty$;

$\zeta = \zeta(z, \mu)$;

$\zeta(\alpha) = \|u(\alpha)^{-1}\|_\infty$;

ζ_* : a designated number used in subproblem (3.8);

$\delta_*, \tau \geq 0$: $\delta(z, \mu) \leq \delta_* = O(n^\tau)$;

$\tilde{\alpha}$: a selected step length defined by (4.10) or (4.11);

$\omega = \max\{1/2, \tau\}$.

3. Algorithms. The purpose of this paper is to study a type of interior point algorithms based on large neighborhoods for solving the following linear complementarity problem:

find $(x, s) \in R^{2n}$ satisfying

$$\begin{aligned} (LCP) \quad & s = Mx + b, \\ & xs = 0, \\ & x, s \geq 0, \end{aligned}$$

where $M \in R^{n \times n}$ and $b \in R^n$ are given matrix and vector. For a survey of interior point methods for (LCP), the reader is referred to the excellent paper [9] by Kojima et al.

Throughout the paper, we make the following assumptions:

A1. Matrix M is positive semidefinite; i.e., matrix M satisfies $\langle x, (Mx) \rangle \geq 0$ for every $x \in R^n$. Note that skew-symmetric matrix, i.e., matrix M satisfying $\langle x, (Mx) \rangle = 0$ for every $x \in R^n$, is included as a special case.

A2. The feasible region of (LCP) has a nonempty interior; i.e., there exists $(x, s) > 0$ such that $s = Mx + b$.

We write $z = (x, s)$. A point z is said to be *strictly feasible* if it is feasible to (LCP) and $z > 0$.

By relaxing the complementary slackness $xs = 0$, we obtain the following system of equations

$$\begin{aligned} (CT)_\mu \quad & xs = \mu e, \\ & s = Mx + b, \\ & x, s \geq 0, \end{aligned}$$

where $\mu > 0$ is a parameter. The solutions to $(CT)_\mu$ define a curve $z^c(\mu) = (x^c(\mu), s^c(\mu))$ with parameter μ called *central trajectory*. For more details about the central trajectory, the reader may be referred to [9], [17], and [21].

For any $\mu > 0$ and any point (x, s) , we define the Newton direction (d_x, d_s) at (x, s) aiming at the point $(x^c(\mu), s^c(\mu))$ as follows:

$$(3.1) \quad \begin{aligned} sd_x + xd_s &= \mu e - xs, \\ d_s &= Md_x. \end{aligned}$$

In our algorithms, the function $\delta(z, \mu)$ defined below is used to measure the *distance* from a point $z = (x, s)$ to the point $z^c(\mu) = (x^c(\mu), s^c(\mu))$ on the central trajectory at μ :

$$(3.2) \quad \delta(z, \mu) = \|u - u^{-1}\|_2, \quad u = \left(\frac{xs}{\mu}\right)^{1/2}.$$

This distance was first introduced to interior point methods by Jansen et al. [5]. A significant feature of this distance is that it tends to infinity as some component $x_i s_i / \mu \rightarrow 0$ or $\rightarrow \infty$; that is, the distance tends to infinity if and only if the point (x, s) tends to the boundary of the feasible region.

Now we present the first-order algorithm.

ALGORITHM 3.1 (first-order algorithm). Given $\epsilon > 0$, $\tau \geq 0$ and $\delta_* = O(n^\tau)$. Let $z^{[0]} = (x^{[0]}, s^{[0]})$ be a strictly feasible solution of (LCP) and $\mu_0 \in (0, \langle x^{[0]}, s^{[0]} \rangle / n)$ (e.g., $\mu_0 = 0.1 \langle x^{[0]}, s^{[0]} \rangle / n$) be a parameter which satisfy

$$\delta(z^{[0]}, \mu_0) \leq \delta_*.$$

Let $k = 0$. Implement the following steps.

Step 1. Calculate the Newton direction $d_z = (d_x, d_s)$ at $(z^{[k]}, \mu_k)$ by (3.1). Set $z^{[k+1]} = z^{[k]} + \alpha_k d_z$, where α_k minimizes $\delta(z^{[k]} + \alpha d_z, \mu_k)$ for $\alpha \geq 0$.

Step 2. Determine μ_{k+1} as the smallest $\mu \in (0, \mu_k)$ which satisfies

$$(3.3) \quad \delta(z^{[k+1]}, \mu) \leq \delta_*.$$

If $\langle x^{[k+1]}, s^{[k+1]} \rangle \leq \epsilon$, stop; $(x^{[k+1]}, s^{[k+1]})$ is an approximate solution. Otherwise, set $k = k + 1$, and go to Step 1.

Remark. As a point tends to the boundary of the feasible region, the distance tends to infinity. Hence the minimum α_k in Step 1 exists and point $z^{[k+1]}$ is strictly feasible.

In the implementation, as well as in the complexity analysis, the step-length α_k is not necessary to be the exact minimum. This will become clear later in sections 4 and 5. Although the use of inexact line search to determine α_k does not incur essential difficulties in practice and theory, it makes the analysis more involved. Therefore, we use the exact minimum α_k in Step 1 of the algorithm for convenience.

If all components of u and u^{-1} are of size $O(1)$, then obviously $\delta(z, \mu) \leq \delta_*$ for some $\delta_* = O(\sqrt{n})$. But $\delta(z, \mu) \leq \delta_*$ also allows some components of u and u^{-1} to be as large as $O(\sqrt{n})$. So the neighborhood implicitly imposed in Algorithm 3.1 (for $\tau = 1/2$) may be larger than the generally used neighborhood with $\beta = O(1)$:

$$\mathcal{N}_\infty^-(\beta) = \left\{ (x, s) \mid \frac{nx s}{\langle x, s \rangle} \geq \beta^{-1} e \right\}$$

(cf. [4], [8], and [12]). We will show that the iteration complexity of Algorithm 3.1 is bounded by $O(n \ln(\mu_0/\epsilon))$ (for $\tau = 1/2$), which is the same complexity bound as achieved for the algorithms using the neighborhood $\mathcal{N}_\infty^-(\beta)$.

The second aim of the paper is to extend Algorithm 3.1 to a high-order algorithm (Algorithm 3.2). The idea of introducing high-order power series in the following way is due to Monteiro, Adler, and Resende [13]. We will show that by using the high-order power series, the upper bound of the iteration complexity can be reduced.

The power series used in the high-order algorithm is an approximate to the trajectory linking the current point, (x, s) , say, and a target point $(x^c(\mu), s^c(\mu))$ on the central trajectory at $\mu > 0$. To construct this trajectory, we observe the following equation which defines the Newton direction (d_x, d_s) at any point $(\bar{x}, \bar{s})$ aiming at the point $(x^c(\mu), s^c(\mu))$:

$$\bar{s}d_x + \bar{x}d_s = \mu e - \bar{x}\bar{s}.$$

A trajectory $(x(\bar{\alpha}), s(\bar{\alpha}))$ whose tangent constantly coincides with the Newton direction is then defined by the following differential equation:

$$\bar{s} \frac{d\bar{x}}{d\bar{\alpha}} + \bar{x} \frac{d\bar{s}}{d\bar{\alpha}} = \mu e - \bar{x}\bar{s}.$$

The above equation can be rewritten as

$$\frac{d(\bar{x}\bar{s})}{d\bar{\alpha}} = \mu e - \bar{x}\bar{s}.$$

The solution of this equation with the initial value $(x(0), s(0)) = (x, s)$ is $(x(\bar{\alpha}), s(\bar{\alpha}))$ satisfying

$$x(\bar{\alpha})s(\bar{\alpha}) = \mu e - (\mu e - xs) \exp(-\bar{\alpha}).$$

Since $\lim_{\bar{\alpha} \rightarrow \infty} x(\bar{\alpha})s(\bar{\alpha}) = \mu e$, the trajectory defined by the above equation converges to the point $(x^c(\mu), s^c(\mu))$. Using the parameter transformation $\alpha = 1 - \exp(-\bar{\alpha})$, we get

$$(3.4) \quad x(\alpha)s(\alpha) = xs + (\mu e - xs)\alpha,$$

with $(x(0), s(0)) = (x, s)$ and $(x(1), s(1)) = (x^c(\mu), s^c(\mu))$. To determine the derivatives of the trajectory, we write $(x(\alpha), s(\alpha))$ as power series:

$$x(\alpha) = x + \sum_{i=1}^{\infty} d_x^{(i)} \alpha^i, \quad s(\alpha) = s + \sum_{i=1}^{\infty} d_s^{(i)} \alpha^i,$$

where $(d_x^{(i)}, d_s^{(i)})$ is the weighted i th derivative. Substituting the above expansions into (3.4), and comparing coefficients of each power of α , we obtain recursive formulas as follows:

$$(3.5) \quad \begin{aligned} s d_x^{(1)} + x d_s^{(1)} &= \mu e - x s, \\ s d_x^{(k)} + x d_s^{(k)} &= - \sum_{i=1}^{k-1} d_x^{(i)} d_s^{(k-i)} \quad \text{for } k = 2, 3, \dots \end{aligned}$$

For the feasibility, the following equations should also be satisfied:

$$(3.6) \quad d_s^{(i)} = M d_x^{(i)} \quad \text{for } i = 1, 2, \dots$$

The high-order algorithm presented in this paper uses the derivatives up to order ν determined by (3.5) and (3.6) to construct a truncated power series to approximate the point $(x^c(\mu), s^c(\mu))$. In the high-order algorithm, we need the following function of $z = (x, s)$ and $\mu > 0$:

$$(3.7) \quad \zeta(z, \mu) = \|u^{-1}\|_{\infty}, \quad u = \left(\frac{x s}{\mu} \right)^{1/2}.$$

ALGORITHM 3.2 (high-order algorithm). Given $\epsilon > 0$, $\tau \geq 0$, $\delta_* = O(n^\tau)$, and $\zeta_* > 0$ (ζ_* may depend on n). Given a strictly feasible solution $z^{[0]}$ and a parameter $\mu_0 \in (0, \langle x^{[0]}, s^{[0]} \rangle / n)$ (e.g., $\mu_0 = 0.1 \langle x^{[0]}, s^{[0]} \rangle / n$) satisfying $\delta(z^{[0]}, \mu_0) \leq \delta_*$ and $\zeta(z^{[0]}, \mu_0) \leq \zeta_*$.

Let $k = 0$. Implement the following steps.

Step 1. Calculate $d_z^{(i)} := (d_x^{(i)}, d_s^{(i)})$, $i = 1, 2, \dots, \nu$, at $(z^{[k]}, \mu_k)$ by (3.5) and (3.6). Set $z^{[k+1]} = z^{[k]} + \sum_{i=1}^{\nu} d_z^{(i)} \alpha_k^i$, where α_k is a solution to the following subproblem:

$$(3.8) \quad \begin{aligned} \min \quad & \delta \left(z^{[k]} + \sum_{i=1}^{\nu} d_z^{(i)} \alpha^i, \mu_k \right) \\ \text{s. t.} \quad & \zeta \left(z^{[k]} + \sum_{i=1}^{\nu} d_z^{(i)} \alpha^i, \mu_k \right) \leq \zeta_*, \\ & \alpha \geq 0. \end{aligned}$$

Step 2. Determine μ_{k+1} as the smallest $\mu \in (0, \mu_k)$ which satisfies

$$(3.9) \quad \delta(z^{[k+1]}, \mu) \leq \delta_*.$$

If $\langle x^{[k+1]}, s^{[k+1]} \rangle \leq \epsilon$, stop; $z^{[k+1]}$ is an approximate solution. Otherwise, set $k = k + 1$, and go to Step 1.

Remark. By (3.7) we see that for any $0 < \mu' \leq \mu$, $\zeta(z, \mu') \leq \zeta(z, \mu)$. Since $\zeta(z^{[k+1]}, \mu_k) \leq \zeta_*$ holds as required by the constraint of (3.8) and $\mu_{k+1} < \mu_k$, we have $\zeta(z^{[k+1]}, \mu_{k+1}) \leq \zeta_*$ for all $k = 0, 1, 2, \dots$

4. Preliminary lemmas. The main work in the complexity analysis of the algorithms is the estimation of the reduction of $\delta(z, \mu_k)$ when z moves from $z^{[k]}$ to $z^{[k+1]} = z^{[k]} + \sum_{i=1}^{\nu} d_z^{(i)} \alpha_k^i$. Because the exact value of the step-length α_k is not known, we will find a lower bound $\tilde{\alpha}$ of α_k such that

$$\delta(z(\tilde{\alpha}), \mu_k) \leq \delta(z^{[k]}, \mu_k) - \sigma,$$

where σ is some positive number and $z(\alpha) = z^{[k]} + \sum_{i=1}^{\nu} d_z^{(i)} \alpha^i$. In this way we obtain a desired estimation as follows:

$$\delta(z^{[k+1]}, \mu_k) \leq \delta(z^{[k]}, \mu_k) - \sigma,$$

since $z^{[k+1]} = z(\alpha_k)$ minimizes the function $\delta(z(\alpha), \mu_k)$ (certainly, $\tilde{\alpha}$ should satisfy the constraints in (3.8) if the high-order algorithm is concerned). The estimation of $\delta(z(\tilde{\alpha}), \mu_k)$ is shown in Lemma 4.5. Lemmas 4.1–4.4 are prepared for Lemma 4.5. Once σ is determined, we can further estimate the reduction on μ ; i.e., estimate a reduction factor $\rho \in (0, 1)$ such that $\mu_{k+1} \leq (1 - \rho)\mu_k$ (see Lemma 4.7). Lemmas 4.5 and 4.7, together with the well-known relation between the duality gap and the parameter μ (see Lemma 4.8), provide the basic material for analyzing the complexity of the algorithms (see section 5).

In this section, we only consider a typical iteration. That is, the index k for the iteration is fixed, and the variation of every quantity only depends on the step-length α . Hence, in this section, we omit the subscript (or superscript) k for the iteration and write every quantity as a function of α . For example,

$$(4.1) \quad u(\alpha) = \left(\frac{(x + \sum_{i=1}^{\nu} d_x^{(i)} \alpha^i)(s + \sum_{i=1}^{\nu} d_s^{(i)} \alpha^i)}{\mu} \right)^{1/2},$$

$$(4.2) \quad \delta(\alpha) = \|u(\alpha) - u(\alpha)^{-1}\|_2, \quad \text{and} \quad \zeta(\alpha) = \|u(\alpha)^{-1}\|_{\infty}.$$

We also define

$$u = u(0) = \left(\frac{xs}{\mu} \right)^{1/2}, \quad h_k = \mu^{-1} \sum_{i=k-\nu}^{\nu} d_x^{(i)} d_s^{(k-i)}.$$

Using (3.5) and after some calculations we can write (cf. [4])

$$(4.3) \quad u(\alpha)^2 = w(\alpha)^2 + r(\alpha),$$

where

$$(4.4) \quad w(\alpha)^2 = u^2 - (u^2 - e)\alpha \quad \text{and} \quad r(\alpha) = \sum_{k=\nu+1}^{2\nu} h_k \alpha^k.$$

In what follows, $|x|$ denotes the vector whose components are the absolute value of components of the vector x .

LEMMA 4.1. *For any $\alpha \in [0, 1]$ such that $(1 - \sqrt{5})/2 \leq w(\alpha)_i^{-2} r(\alpha)_i \leq 1 \ \forall i = 1, \dots, n$, the following inequality holds true:*

$$|u(\alpha) - u(\alpha)^{-1}| \leq |w(\alpha) - w(\alpha)^{-1}| + (w(\alpha)^{-1} + w(\alpha)^{-3})|r(\alpha)|.$$

Proof. For any $\lambda \in R$ with $(1 - \sqrt{5})/2 \leq \lambda \leq 1$, one can verify that

$$|(1 + \lambda)^{1/2} - 1| \leq |\lambda| \quad \text{and} \quad |(1 + \lambda)^{-1/2} - 1| \leq |\lambda|.$$

So we have

$$\begin{aligned} |u(\alpha) - u(\alpha)^{-1}| &\leq |w(\alpha) - w(\alpha)^{-1}| + w(\alpha)|(e + w(\alpha)^{-2}r(\alpha))^{1/2} - e| \\ &\quad + w(\alpha)^{-1}|(e + w(\alpha)^{-2}r(\alpha))^{-1/2} - e| \\ &\leq |w(\alpha) - w(\alpha)^{-1}| + (w(\alpha)^{-1} + w(\alpha)^{-3})|r(\alpha)|. \quad \square \end{aligned}$$

Let $y \in R^n$ be any vector and $N \subset \{1, 2, \dots, n\}$ be any index set. We denote by y_N the subvector consisting of those components of y whose indices belong to N .

LEMMA 4.2. *For any index set $N \subset \{1, 2, \dots, n\}$ and any $\alpha \in [0, 1]$, we have*

$$(4.5) \quad \|(w(\alpha) - w(\alpha)^{-1})_N\|_2 \leq (1 - \frac{\alpha}{2})\|(u - u^{-1})_N\|_2,$$

$$(4.6) \quad \|w(\alpha)_N^{-1}\|_\infty = \|u_N^{-1}\|_\infty (1 - \alpha + \alpha\|u_N^{-1}\|_\infty^2)^{-1/2}.$$

Proof. One can verify that the function

$$\xi(\alpha) = \|(w(\alpha) - w(\alpha)^{-1})_N\|_2^2$$

is convex, and

$$\xi(0) = \|(u - u^{-1})_N\|_2^2, \quad \xi(1) = 0.$$

This implies that for any $\alpha \in [0, 1]$,

$$\xi(\alpha) \leq \xi(0)(1 - \alpha) + \xi(1)\alpha = \|(u - u^{-1})_N\|_2^2(1 - \alpha).$$

So we have

$$\|(w(\alpha) - w(\alpha)^{-1})_N\|_2 \leq (1 - \alpha)^{1/2}\|(u - u^{-1})_N\|_2 \leq (1 - \frac{\alpha}{2})\|(u - u^{-1})_N\|_2.$$

This proves the inequality (4.5).

To prove (4.6) we use the definition of $w(\alpha)$ in (4.4) and write for every component $i \in N$

$$w(\alpha)_i^{-1} = [u_i^2 - (u_i^2 - 1)\alpha]^{-1/2} = u_i^{-1}(1 - \alpha + \alpha u_i^{-2})^{-1/2}.$$

Because the function $\phi(\lambda) = \lambda(1 - \alpha + \lambda^2\alpha)^{-1/2}$ is monotonely increasing for any fixed $\alpha \in [0, 1]$, the maximum of $w(\alpha)_i^{-1}$ for $i \in N$ must be reached when u_i^{-1} reaches its maximum for $i \in N$. This proves the identity (4.6). $\square$

To estimate $r(\alpha)$, we should first estimate the derivatives $(d_x^{(i)}, d_s^{(i)})$ of the trajectory from a point (x, s) to the point $(x^c(\mu), s^c(\mu))$. We denote

$$D = \text{diag}(x^{-1/2}s^{1/2}), \quad u = \left(\frac{xs}{\mu}\right)^{1/2}.$$

Then (3.5) can be rewritten as

$$(4.7) \quad \begin{aligned} Dd_x^{(1)} + D^{-1}d_s^{(1)} &= -\mu^{1/2}(u - u^{-1}), \\ Dd_x^{(k)} + D^{-1}d_s^{(k)} &= -\mu^{-1/2}u^{-1} \sum_{i=1}^{k-1} d_x^{(i)} d_s^{(k-i)} \quad \text{for } k = 2, 3, \dots \end{aligned}$$

Denote

$$\eta_k = \|Dd_x^{(k)} + D^{-1}d_s^{(k)}\|_2, \quad \zeta = \|u^{-1}\|_\infty, \quad \delta = \|u - u^{-1}\|_2.$$

Because

$$(4.8) \quad \langle Dd_x^{(k)}, D^{-1}d_s^{(k)} \rangle = \langle d_x^{(k)}, d_s^{(k)} \rangle = \langle d_x^{(k)}, (Md_x^{(k)}) \rangle \geq 0,$$

we have

$$\max \{ \|Dd_x^{(k)}\|_2, \|D^{-1}d_s^{(k)}\|_2 \} \leq \eta_k.$$

The lemma below gives an upper bound on η_k (also see [4] and [22]).

LEMMA 4.3.

$$\max \{ \|Dd_x^{(k)}\|_2, \|D^{-1}d_s^{(k)}\|_2 \} \leq \eta_k \leq \theta_k \mu^{1/2} \zeta^{k-1} \delta^k,$$

where

$$\theta_k = \frac{1}{k} \binom{2k-2}{k-1} \leq \frac{1}{k} 4^k$$

is the well-known solution to the following equations:

$$\theta_1 = 1, \quad \theta_k = \sum_{i=1}^{k-1} \theta_i \theta_{k-i}, \quad k = 2, 3, \dots$$

Proof. We use the induction to prove the inequality $\eta_k \leq \theta_k \mu^{1/2} \zeta^{k-1} \delta^k$. Obviously, it holds true for $k = 1$. Now for $k \geq 2$ suppose it holds for all $1 \leq i < k$; then from (4.7) we have

$$\begin{aligned} \eta_k &\leq \mu^{-1/2} \zeta \sum_{i=1}^{k-1} \eta_i \eta_{k-i} \\ &\leq \mu^{-1/2} \zeta \sum_{i=1}^{k-1} \theta_i \theta_{k-i} \mu \zeta^{k-2} \delta^k \\ &= \theta_k \mu^{1/2} \zeta^{k-1} \delta^k. \end{aligned}$$

This shows that the inequality also holds true for k . The proof is completed. $\square$

Using Lemma 4.3 we can estimate h_k for $k = \nu + 1, \nu + 2, \dots, 2\nu$

$$(4.9) \quad \begin{aligned} \|h_k\|_2 &\leq \mu^{-1} \sum_{i=k-\nu}^{\nu} \eta_i \eta_{k-i} \\ &\leq \zeta^{k-2} \delta^k \sum_{i=k-\nu}^{\nu} \theta_i \theta_{k-i} \\ &\leq \theta_k \zeta^{k-2} \delta^k \\ &\leq \frac{(4\zeta\delta)^k}{k\zeta^2}. \end{aligned}$$

Our main task is to estimate how much we can reduce $\delta(z, \mu_k)$ when we move from $z = z^{[k]}$ to $z = z^{[k+1]}$. That is, we should estimate an amount $\sigma > 0$ such that $\delta(\alpha_k) \leq \delta - \sigma$. Since for any $\alpha \geq 0$ (subject to the constraint of the subproblem (3.8) if Algorithm 3.2 is concerned), $\delta(\alpha_k) \leq \delta(\alpha)$, we need only to choose a suitable $\tilde{\alpha}$ and establish the estimate $\delta(\tilde{\alpha}) \leq \delta - \sigma$. We make the following choice of step-length $\tilde{\alpha}$ for the complexity analysis (but not for implementation of the algorithm). For the first order algorithm, i.e., $\nu = 1$, we let

$$(4.10) \quad \tilde{\alpha} = \frac{1}{48\delta_*^2}.$$

For the high-order algorithm, i.e., $\nu \geq 2$, we let

$$(4.11) \quad \tilde{\alpha} = \frac{1}{3 \cdot 4^{\frac{\nu+1}{\nu}} \zeta_*^{\frac{\nu-1}{\nu}} \delta_*^{\frac{\nu+1}{\nu}}}.$$

Note that $\tilde{\alpha}$ in (4.10) is the special form of $\tilde{\alpha}$ in (4.11) by setting $\nu = 1$. Hence, in what follows we sometimes only consider $\tilde{\alpha}$ in (4.11). Without loss of generality, in what follows we always assume that

$$2 \leq \zeta_* \leq \delta_*.$$

Now we estimate some quantities related to the step-length $\tilde{\alpha}$.

LEMMA 4.4. *For the vector $r(\alpha)$ defined by (4.4) and the step-length $\tilde{\alpha}$ defined by (4.10) or (4.11), we have*

$$\|r(\tilde{\alpha})\|_2 \leq \frac{\tilde{\alpha}}{6}.$$

Proof. Using (4.9) we have

$$\|r(\tilde{\alpha})\|_2 \leq \sum_{k=\nu+1}^{2\nu} \|h_k\|_2 \tilde{\alpha}^k \leq \frac{1}{(\nu+1)\zeta^2} \sum_{k=\nu+1}^{2\nu} (4\zeta\delta\tilde{\alpha})^k.$$

For $\nu = 1$, we have by (4.10)

$$\|r(\tilde{\alpha})\|_2 \leq \frac{1}{2}(4\delta\tilde{\alpha})^2 \leq \frac{\tilde{\alpha}}{6}.$$

For $\nu \geq 2$, noting that $\zeta \leq \zeta_*$ and $\delta \leq \delta_*$, we have by (4.11)

$$4\zeta\delta\tilde{\alpha} \leq \frac{1}{3} \left(\frac{\zeta_*}{\delta_*} \right)^{1/\nu} \leq \frac{1}{3}.$$

The last inequality uses the assumption $\zeta_* \leq \delta_*$. Finally, we have

$$\|r(\tilde{\alpha})\|_2 \leq \frac{(4\zeta\delta\tilde{\alpha})^{\nu+1}}{(\nu+1)\zeta^2(1-4\zeta\delta\tilde{\alpha})} \leq \frac{\tilde{\alpha}}{3^{\nu}(\nu+1)(1-4\zeta\delta\tilde{\alpha})} \leq \frac{\tilde{\alpha}}{6}. \quad \square$$

We are ready to estimate an upper bound on $\delta(\tilde{\alpha})$.

LEMMA 4.5. *For the step-length $\tilde{\alpha}$ defined by (4.10) or (4.11), the following inequality holds:*

$$\delta(\tilde{\alpha}) \leq \delta - \frac{\tilde{\alpha}}{4}\delta + \frac{5\tilde{\alpha}}{3}.$$

Proof. We define two index sets for the given u as follows:

$$L = \{i \mid 0 < u_i < 1/2\} \quad \text{and} \quad S = \{i \mid u_i \geq 1/2\}.$$

Let p be any vector, then p_L and p_S denote the subvectors consisting of those components of p whose indices belong to L and S , respectively.

Using the notation $w(\tilde{\alpha}) = u(e + (u^{-2} - e)\tilde{\alpha})^{1/2}$, and noticing that $w(\tilde{\alpha})_L^{-1} \leq u_L^{-1}$, we have

$$\begin{aligned} (u^{-1} - w(\tilde{\alpha})^{-1})_L &= w(\tilde{\alpha})_L^{-1} \left((e + (u^{-2} - e)\tilde{\alpha})^{1/2} - e \right)_L \\ &= w(\tilde{\alpha})_L^{-1} \left((e + (u^{-2} - e)\tilde{\alpha})^{1/2} + e \right)_L^{-1} (u^{-2} - e)_L \tilde{\alpha} \\ &\geq \frac{1}{2} w(\tilde{\alpha})_L^{-1} (e + (u^{-2} - e)\tilde{\alpha})_L^{-1/2} (u^{-2} - e)_L \tilde{\alpha} \\ &= \frac{1}{2} w(\tilde{\alpha})_L^{-2} (u^{-1} - u)_L \tilde{\alpha} \\ &\geq \frac{1}{2} w(\tilde{\alpha})_L^{-3} (e - u^2)_L \tilde{\alpha}. \end{aligned}$$

Similarly, we have

$$(w(\tilde{\alpha}) - u)_L \geq \frac{1}{2} w(\tilde{\alpha})_L^{-1} (e - u^2)_L \tilde{\alpha}.$$

It follows from Lemma 4.4 and $u_i < 1/2$ that

$$|r(\tilde{\alpha})_L| \leq \|r(\tilde{\alpha})\|_2 e \leq \frac{\tilde{\alpha}}{6} e \leq \frac{\tilde{\alpha}}{4} (e - u^2)_L.$$

Hence

$$\begin{aligned} (w(\tilde{\alpha})^{-1} + w(\tilde{\alpha})^{-3})_L |r(\tilde{\alpha})_L| &\leq \frac{1}{4} (w(\tilde{\alpha})^{-1} + w(\tilde{\alpha})^{-3})_L (e - u^2)_L \tilde{\alpha} \\ &\leq \frac{1}{2} [(u^{-1} - w(\tilde{\alpha})^{-1}) + (w(\tilde{\alpha}) - u)]_L. \end{aligned}$$

Since for every $i \in L$, $w(\tilde{\alpha})_i^{-1} \leq u_i^{-1} \leq \zeta_* \leq \delta_*$, it is easy to verify, by using Lemma 4.4, that the condition of Lemma 4.1 is satisfied. It follows from Lemma 4.1 and the above inequality that

$$\begin{aligned} |(u(\tilde{\alpha})^{-1} - u(\tilde{\alpha}))_L| &\leq (w(\tilde{\alpha})^{-1} - w(\tilde{\alpha}))_L + (w(\tilde{\alpha})^{-1} + w(\tilde{\alpha})^{-3})_L |r(\tilde{\alpha})_L| \\ &\leq \frac{1}{2} (u^{-1} - u)_L + \frac{1}{2} (w(\tilde{\alpha})^{-1} - w(\tilde{\alpha}))_L. \end{aligned}$$

By using (4.5) and denoting $q = u - u^{-1}$ and $q(\alpha) = u(\alpha) - u(\alpha)^{-1}$, we have

$$\|q(\tilde{\alpha})_L\|_2 \leq \frac{1}{2} \|q_L\|_2 + \frac{1}{2} (1 - \frac{\tilde{\alpha}}{2}) \|q_L\|_2 = (1 - \frac{\tilde{\alpha}}{4}) \|q_L\|_2.$$

For the index set S we have from (4.6) and $\|u_S^{-1}\|_\infty \leq 2$ that $\|w(\tilde{\alpha})_S^{-1}\|_\infty \leq 2$. Hence the condition of Lemma 4.1 is also satisfied. Then it follows from Lemma 4.1 that

$$\begin{aligned} \|q(\tilde{\alpha})_S\|_2 &\leq \|(w(\tilde{\alpha}) - w(\tilde{\alpha})^{-1})_S\|_2 + \|(w(\tilde{\alpha})^{-1} + w(\tilde{\alpha})^{-3})_S\|_\infty \|r(\tilde{\alpha})\|_2 \\ &\leq \|(w(\tilde{\alpha}) - w(\tilde{\alpha})^{-1})_S\|_2 + 10 \|r(\tilde{\alpha})\|_2. \end{aligned}$$

Then by using (4.5) and Lemma 4.4, we have

$$\|q(\tilde{\alpha})_S\|_2 \leq (1 - \frac{\tilde{\alpha}}{2})\|q_S\|_2 + \frac{5\tilde{\alpha}}{3}.$$

Now we sum up the inequalities for L and S below.

$$\begin{aligned} \|q(\tilde{\alpha})\|_2 &= \sqrt{\|q(\tilde{\alpha})_L\|_2^2 + \|q(\tilde{\alpha})_S\|_2^2} \\ &\leq \sqrt{(1 - \frac{\tilde{\alpha}}{4})^2 \|q_L\|_2^2 + ((1 - \frac{\tilde{\alpha}}{4})\|q_S\|_2 + \frac{5\tilde{\alpha}}{3})^2} \\ &\leq \sqrt{(1 - \frac{\tilde{\alpha}}{4})^2 \|q\|_2^2 + \frac{10\tilde{\alpha}}{3}(1 - \frac{\tilde{\alpha}}{4})\|q_S\|_2 + (\frac{5\tilde{\alpha}}{3})^2} \\ &\leq (1 - \frac{\tilde{\alpha}}{4})\|q\|_2 + \frac{5\tilde{\alpha}}{3}. \quad \square \end{aligned}$$

To analyze the complexity of Algorithm 3.2, we still need to show that the step length chosen as (4.11) satisfies the constraints in subproblem (3.8).

LEMMA 4.6. *Let $\zeta_* \geq 2$ be any number. If $\zeta = \zeta(z, \mu) \leq \zeta_*$, then for $\tilde{\alpha}$ defined by (4.11) we have*

$$\zeta(\tilde{\alpha}) = \zeta\left(z + \sum_{i=1}^{\nu} d_z^{(i)} \tilde{\alpha}^i, \mu\right) \leq \zeta_*.$$

Proof. Similar to Lemma 4.1, we can derive

$$u(\tilde{\alpha})^{-1} \leq w(\tilde{\alpha})^{-1}(1 + w(\tilde{\alpha})^{-2}|r(\tilde{\alpha})|).$$

Using (4.6) and noting that $\phi(\zeta) = \zeta(1 - \alpha + \zeta^2\alpha)^{1/2}$ is monotonely increasing for any fixed $\alpha \in [0, 1]$, we have

$$\|w(\tilde{\alpha})^{-1}\|_{\infty} = \zeta(1 - \tilde{\alpha} + \zeta^2\tilde{\alpha})^{-1/2} \leq \zeta_*(1 - \tilde{\alpha} + \zeta_*^2\tilde{\alpha})^{-1/2} \leq \zeta_*.$$

Then using Lemma 4.4 we derive

$$\zeta(\tilde{\alpha}) \leq \zeta_* \frac{1 + \tilde{\alpha}\zeta_*^2/6}{[1 + \tilde{\alpha}(\zeta_*^2 - 1)]^{1/2}}.$$

Using (4.11) and $\zeta_* \leq \delta_*$, one can verify that $(\zeta_*^2 - 1)\tilde{\alpha} \leq 1/12$. Hence

$$(1 + (\zeta_*^2 - 1)\tilde{\alpha})^{1/2} \geq 1 + 0.48(\zeta_*^2 - 1)\tilde{\alpha}.$$

For $\zeta_* \geq 2$, we have $0.48(\zeta_*^2 - 1) \geq \zeta_*^2/6$. This implies $\zeta(\tilde{\alpha}) \leq \zeta_*$. $\square$

The following lemma relates the reduction on the parameter μ to the reduction on the distance δ .

LEMMA 4.7. *For any point $\hat{z} = (\hat{x}, \hat{s}) > 0$ and $\mu > 0$, suppose that*

$$\delta(\hat{z}, \mu) \leq \delta_* - \sigma$$

for some number σ with $0 < \sigma \leq \delta^*$. Let

$$\hat{\mu} = (1 - \rho)\mu, \quad \rho = \frac{\sigma}{\delta_* + \sqrt{n}} < 1.$$

Then the following inequality holds:

$$\delta(\hat{z}, \hat{\mu}) \leq \delta_*.$$

Proof. Write

$$\bar{u} = \left(\frac{\hat{x}\hat{s}}{\mu}\right)^{1/2}, \quad \hat{u} = \left(\frac{\hat{x}\hat{s}}{\hat{\mu}}\right)^{1/2}.$$

For any n -vector $\xi > 0$ the following inequality holds:

$$(4.12) \quad \|\xi\|_2 \leq \| |\xi - \xi^{-1}| + e \|_2 \leq \|\xi - \xi^{-1}\|_2 + \sqrt{n}.$$

So we have

$$\begin{aligned} \delta(\hat{z}, \hat{\mu}) &= \|\hat{u} - \hat{u}^{-1}\|_2 \\ &= \|(1 - \rho)^{-1/2}\bar{u} - (1 - \rho)^{1/2}\bar{u}^{-1}\|_2 \\ &= \|\rho(1 - \rho)^{-1/2}\bar{u} + (1 - \rho)^{1/2}(\bar{u} - \bar{u}^{-1})\|_2 \\ &\leq \rho(1 - \rho)^{-1/2}[\delta(\hat{z}, \mu) + \sqrt{n}] + (1 - \rho)^{1/2}\delta(\hat{z}, \mu) \quad \text{using (4.12)} \\ &= (1 - \rho)^{-1/2}[\delta(\hat{z}, \mu) + \rho\sqrt{n}] \\ &\leq (1 - \rho)^{-1/2}[\delta_* - \sigma + \rho\sqrt{n}] \\ &= (1 - \rho)^{-1/2}(1 - \rho)\delta_* \\ &\leq \delta_*. \quad \square \end{aligned}$$

The stopping criterion in the algorithms is imposed on the duality gap $\langle x, s \rangle$, while at each iteration only the parameter μ is reduced. A relation between them should be shown, so we have the following lemma which is essentially the same as Lemma 3.1 in [16].

LEMMA 4.8. *Let $\tau \geq 0$ be any real number. If $\delta = \delta(x, s, \mu) \leq \delta_*$, then*

$$\langle x, s \rangle \leq (\delta_*^2 + \delta_* n^{1/2} + n)\mu.$$

Proof. Using the Cauchy-Schwarz inequality, we have

$$\frac{\langle x, s \rangle}{\mu} - \langle e, e \rangle = \langle u, (u - u^{-1}) \rangle \leq \|u\|_2 \|u - u^{-1}\|_2.$$

It follows then from (4.12) that

$$\frac{\langle x, s \rangle}{\mu} - \langle e, e \rangle \leq (\delta + \sqrt{n})\delta \leq \delta_*^2 + \delta_* n^{1/2}.$$

The inequality of the lemma follows immediately since $\langle e, e \rangle = n$. $\square$

5. Complexity. In this section we present in two theorems the complexity bounds for the two algorithms. Since we are interested in large-step algorithms, we assume $\delta_* \geq 7$. For the first-order algorithm (Algorithm 3.1) we have the following theorem.

THEOREM 5.1. *Let $\delta_* = O(n^\tau) \geq 7$, $\tau \geq 0$. Then Algorithm 3.1 needs at most $O(n^{\tau+\omega} \ln(\mu_0 n^{2\omega}/\epsilon))$ iterations to find a feasible point (x, s) with $\langle x, s \rangle \leq \epsilon$, where $\omega = \max\{1/2, \tau\}$.*

Proof. Let the sequence of $(z^{[k]}, \mu_k)$, $k = 0, 1, 2, \dots$, be generated by Algorithm 3.1. Noting that α_k minimizes $\delta(\alpha)$ for $\alpha \geq 0$, we have

$$\delta(z^{[k+1]}, \mu_k) = \delta(\alpha_k) \leq \delta(\alpha) \quad \forall \alpha \geq 0.$$

Let $\tilde{\alpha}$ be the number given by (4.10); that is, $\tilde{\alpha} = 1/(48\delta_*^2)$. By Lemma 4.5 we have

$$\delta(z^{[k+1]}, \mu_k) \leq \delta(\tilde{\alpha}) \leq \delta_* - \tilde{\alpha}\delta_*\left(\frac{1}{4} - \frac{5}{3\delta_*}\right) \leq \delta_* - \frac{C_1}{n^\tau},$$

where C_1 is a finite positive constant since we assume $\delta_* \geq 7$. Let

$$(5.1) \quad \hat{\mu} = (1 - \rho)\mu_k, \quad \rho = \frac{C_1}{n^\tau(\delta_* + \sqrt{n})} < 1.$$

By Lemma 4.7,

$$\delta(z^{[k+1]}, \hat{\mu}) \leq \delta_*.$$

This shows that $\hat{\mu}$ satisfies condition (3.3) and implies that

$$\mu_{k+1} \leq \hat{\mu} = (1 - \rho)\mu_k.$$

Hence

$$(5.2) \quad \mu_k \leq (1 - \rho)^k \mu_0.$$

By Lemma 4.8, $\langle x^{[k]}, s^{[k]} \rangle \leq \epsilon$ is achieved if

$$(5.3) \quad \mu_k \leq \frac{\epsilon}{\delta_*^2 + \delta_* n^{1/2} + n}.$$

Obviously, there exists a constant $C_2 > 0$ such that

$$\frac{C_2\epsilon}{n^{2\omega}} \leq \frac{\epsilon}{\delta_*^2 + \delta_* n^{1/2} + n}.$$

Then in order to achieve (5.3), by (5.2) k needs only to satisfy

$$(1 - \rho)^k \mu_0 \leq \frac{C_2\epsilon}{n^{2\omega}};$$

that is,

$$k \ln(1 - \rho) \leq \ln\left(\frac{C_2\epsilon}{\mu_0 n^{2\omega}}\right).$$

Since $\ln(1 - \rho) \approx -\rho$ for small ρ , the above inequality is equivalent to

$$k \geq \rho^{-1} \ln\left(\frac{\mu_0 n^{2\omega}}{C_2\epsilon}\right).$$

Summarizing the above and noting that $\rho^{-1} = O(n^{\tau+\omega})$ we complete the proof. $\square$

Remark. If $\tau = 1/2$, the complexity bound is $O(n \ln(\mu_0 n / \epsilon))$. This is the best complexity bound to date for large-step algorithms. Next, let us observe the power of n in the complexity upper bound (omitting the $\ln$ -term), which is $O(n^{2\tau})$ for $\tau \geq 1/2$.

This shows that the complexity is a polynomial of n with its degree linearly depending on the radius of the neighborhood, $O(n^\tau)$. As $\tau \rightarrow \infty$, the complexity tends to an exponential one, $O(e^n)$. So we can imagine that a large-neighborhood algorithm as $\tau \rightarrow \infty$ should behave as the simplex method whose worst-case complexity is known to be exponential. The observation on numerical and theoretical results seems to support this conjecture.

The following theorem presents an upper bound for the iteration complexity of the high-order algorithm (Algorithm 3.2).

THEOREM 5.2. *Let $\delta_* = O(n^\tau) \geq 7$, $\tau \geq 0$, and let $\zeta_* \geq 2$ be any number (ζ_* may depend on n). Then Algorithm 3.2 needs at most $O(\zeta_*^{\frac{\nu-1}{\nu}} n^{\frac{\tau}{\nu} + \omega} \ln(\mu_0 n^{2\omega}/\epsilon))$ iterations to find a feasible solution (x, s) with $\langle x, s \rangle \leq \epsilon$.*

Proof. Let $\tilde{\alpha}$ be the number given by (4.11). The remark to Algorithm 3.2 indicates that $\zeta(z^{[k]}, \mu_k) \leq \zeta_*$, so by Lemma 4.6 $\tilde{\alpha}$ satisfies the constraints in subproblem (3.8). Therefore,

$$\delta(z^{[k+1]}, \mu_k) \leq \delta(\tilde{\alpha}).$$

By Lemma 4.5 we have

$$\delta(\tilde{\alpha}) \leq \delta_* - \tilde{\alpha} \delta_* \left(\frac{1}{4} - \frac{5}{3\delta_*} \right).$$

Noting that $\tilde{\alpha} \delta_* \geq C_0 / (\zeta_*^{\frac{\nu-1}{\nu}} n^{\frac{\tau}{\nu}})$ for some constant $C_0 > 0$ and that we assume $\delta_* \geq 7$, there exists a finite positive number C_1 such that

$$\delta(z^{[k+1]}, \mu_k) \leq \delta_* - \frac{C_1}{\zeta_*^{\frac{\nu-1}{\nu}} n^{\frac{\tau}{\nu}}}.$$

Let

$$\rho = \frac{C_1}{\zeta_*^{\frac{\nu-1}{\nu}} n^{\frac{\tau}{\nu}} (\delta_* + \sqrt{n})}.$$

Then the remaining proof is an analogue to the proof of Theorem 5.1. $\square$

COROLLARY 5.3. *Let $\nu = \lceil n^\gamma \rceil$ with some positive number γ as the order of Algorithm 3.2. Let ζ_* and δ_* be as in Theorem 5.2. Then Algorithm 3.2 needs at most $O(\zeta_*^{\frac{\nu-1}{\nu}} n^\omega \ln(\mu_0 n^{2\omega}/\epsilon))$ iterations to find a feasible solution (x, s) with $\langle x, s \rangle \leq \epsilon$. The “O” in the complexity bound depends on γ in the form of $e^{\frac{\tau}{\gamma\epsilon}}$.*

Proof. We need only to show that

$$g(n) = n^{\tau/\nu} = n^{\tau/n^\gamma}$$

is bounded for all $n \geq 1$.

First we notice that

$$\lim_{n \rightarrow \infty} g(n) = 1,$$

since $\gamma > 0$. So we know that continuous function $g(n)$ must be bounded. We would like further to find an upper bound for $g(n)$. For this purpose, we need only to differentiate $g(n)$ (here n is viewed as a continuous variable). We find that $n = e^{1/\gamma}$ maximizes $g(n)$. Thus for all $n \geq 1$,

$$g(n) \leq g(e^{1/\gamma}) = e^{\frac{\tau}{\gamma\epsilon}}. \quad \square$$

This corollary is essentially due to C. Roos (see footnote 2 on page 1).

If $\gamma = 0.1$, $\tau = 1/2$, then

$$e^{\frac{\tau}{\gamma\epsilon}} \leq 7.$$

This does not affect the complexity bound. On the other hand, we have a very low order for the algorithm. For instance, to solve a problem of dimension $n = 10^6$, an order of $\nu = 4$ is enough.

We can roughly estimate the arithmetic operations needed by the high-order algorithm. At each iteration $O(n^3)$ arithmetic operations are needed to compute the Newton direction. In the algorithm of order ν we need only $O(n^3 + \nu n^2)$ arithmetic operations to compute the ν derivatives because the same matrix is used for the computation of all derivatives (see formulas (3.5) and (3.6)). Hence each iteration of a high-order algorithm also requires $O(n^3)$ arithmetic operations provided $\nu \leq n$.

If we let $\zeta_* = O(1)$ and let $\nu = \lceil n^\gamma \rceil$ for some $\gamma > 0$ in Algorithm 3.2, the iteration complexity according to Corollary 5.3 is $O(n^\omega \ln(\mu_0 n^{2\omega}/\epsilon))$. For $\tau = 1/2$, it is $O(\sqrt{n} \ln(n\mu_0/\epsilon))$, the well-known best upper bound of the iteration complexity for interior point algorithms. If we choose an appropriate γ , e.g., $\gamma = 0.1$, then the order of the algorithm, $\nu = n^\gamma$, is fairly small even for large-scale problems. So our high-order algorithm enjoys both “low” order and “low” complexity. This result gives a good explanation to the computational results obtained by implementing the high-order algorithms which use two to five derivatives and show a significant reduction on the computing time compared to the first-order algorithms; cf. [11], [7].

Finally, we would like to mention that Algorithm 3.2 is not exactly an extension of Algorithm 3.1 because it has an additional constraint $\zeta(\alpha) \leq \zeta_*$. It is worthwhile to further investigate the algorithm without this constraint.

Acknowledgments. The author wishes to thank C. Roos for his helpful and constructive comments which improve the results in this paper (see footnote 2 on page 1). Thanks are also due to associate editor K. M. Anstreicher and the anonymous referees for their helpful comments.

REFERENCES

- [1] R. M. FREUND, *Polynomial-time algorithms for linear programming based only on primal scaling and projected gradients of a potential function*, Math. Programming, 51 (1991), pp. 203–222.
- [2] C. GONZAGA, *Large step path-following methods for linear programming, part I: Barrier function method*, SIAM J. Optim., 1 (1991), pp. 268–279.
- [3] C. GONZAGA, *Large step path-following methods for linear programming, part II: Potential reduction method*, SIAM J. Optim., 1 (1991), pp. 280–292.
- [4] P. HUNG AND Y. YE, *An Asymptotical $O(\sqrt{n}L)$ -Iteration Path-Following Linear Programming Algorithm that Uses Long Steps*, Technical report no. 53, on Computational Mathematics, Dept. of Mathematics, University of Iowa, Iowa City, IA, 1994.
- [5] B. JANSEN, C. ROOS, T. TERLAKY, AND J.-PH. VIAL, *Primal-Dual Algorithms for Linear Programming Based on the Logarithmic Barrier Method*, Technical report no. 92–104, Faculty of Technical Mathematics and Informatics, Delft University of Technology, The Netherlands, 1992.
- [6] B. JANSEN, C. ROOS, T. TERLAKY, AND J.-PH. VIAL, *Primal-Dual Target-Following Algorithms for Linear Programming*, Technical report no. 93–107, Faculty of Technical Mathematics and Informatics, Delft University of Technology, The Netherlands, 1993.
- [7] N. K. KARMARKAR, J. C. LAGARIAS, L. SLUTSMAN, AND P. WANG, *Power series variants of Karmarkar-type algorithms*, AT&T Technical Journal, 68 (1989), pp. 20–36.
- [8] M. KOJIMA, N. MEGIDDO, AND S. MIZUNO, *Theoretical convergence of large-step primal-dual interior point algorithms for linear programming*, Math. Programming, 59 (1993), pp. 1–21.

- [9] M. KOJIMA, N. MEGIDDO, T. NOMA, AND A. YOSHISE, *A Unified Approach to Interior Point Algorithms for Linear Complementarity Problems*, Lecture Notes in Computer Science 538, Springer-Verlag, Berlin, 1991.
- [10] M. KOJIMA, S. MIZUNO, AND A. YOSHISE, *A primal-dual interior point algorithm for linear programming*, in *Progress in Mathematical Programming: Interior-Point and Related Methods*, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 29–47.
- [11] S. MEHROTRA, *Higher Order Methods and Their Performance*, Technical report 90-16R1, Dept. of Industrial Engineering and Management Science, Northwestern University, Evanston, IL, 1990 (revised July 1991).
- [12] S. MIZUNO, M. J. TODD, AND Y. YE, *On adaptive-step primal-dual interior-point algorithms for linear programming*, *Math. Oper. Res.*, 18 (1993), pp. 964–981.
- [13] R. C. MONTEIRO, I. ADLER, AND M. G. C. RESENDE, *A polynomial-time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extension*, *Math. Oper. Res.*, 15 (1990), pp. 191–214.
- [14] F. A. POTRA, *On a Predictor-Corrector Method for Solving Linear Programs from Infeasible Starting Points*, Technical report 34, Department of Mathematics, The University of Iowa, Iowa City, IA, 1992.
- [15] J. RENEGAR, *A polynomial-time algorithm based on Newton's method for linear programming*, *Math. Programming*, 40 (1988), pp. 59–93.
- [16] C. ROOS AND J.-PH. VIAL, *Long steps with the logarithmic penalty barrier function in linear programming*, in *Economic Decision-Making: Games, Economics and Optimization*, J. Gabszewicz, J. F. Richard, and L. Wolsey, eds., Elsevier, Amsterdam, 1989, pp. 433–441.
- [17] G. SONNEVEND, *An "analytical center" for polyhedrons and new classes for linear (smooth, convex) programming*, in *System Modelling and Optimization (Budapest, 1985)*, Lecture Notes in Control and Inform. Sci. 84, A. Prékopa, J. Szelecsan, and B. Strazicky, eds., Springer-Verlag, New York, 1985, pp. 866–876.
- [18] M. TODD, *Recent developments and new directions in linear programming*, in *Mathematical Programming—Recent Developments and Applications*, M. Iri and K. Tanabe, eds., Kluwer Academic Publishers, Norwell, MA, 1989, pp. 109–157.
- [19] Y. YE, *An $O(n^3L)$ potential reduction algorithm for linear programming*, *Math. Programming*, 50 (1991), pp. 239–258.
- [20] L. ZHANG AND Y. ZHANG, *On Polynomiality of the Mehrotra-Type Predictor-Corrector Interior-Point Algorithms*, Technical report 93–12, Department of Mathematics and Statistics, University of Maryland at Baltimore County, Baltimore, MD, 1993.
- [21] G. ZHAO AND J. STOER, *Estimating the complexity of a class of path-following methods for solving linear programming by curvature integrals*, *Appl. Math. Optim.*, 27 (1993), pp. 85–103.
- [22] G. ZHAO AND J. ZHU, *The curvature integral and the complexity of linear complementarity problems*, *Math. Programming*, 70 (1995), pp. 107–122.
- [23] G. ZHAO, J. SUN, AND J. ZHU, *A primal-dual affine scaling algorithm with necessary centering as a safeguard*, *Optimization*, 35 (1995), pp. 333–343.

EXISTENCE OF SOLUTIONS TO DISCRETE SEMICOERCIVE FRICTIONAL CONTACT PROBLEMS*

ANDERS KLARBRING[†] AND JONG-SHI PANG[‡]

Abstract. This paper establishes new, sufficient conditions for the existence of a solution to the discrete, three-dimensional, deformable body, small displacement, frictional contact problem with a positive semidefinite stiffness matrix. The approach is based on a quasi-variational inequality formulation of the problem and makes extensive use of known results from linear complementarity theory.

Key words. contact problems, Coulomb friction, solution existence, complementarity problems, quasi-variational inequality

AMS subject classifications. 73K12, 73K40, 90C33, 90C90

PII. S105262349629784X

1. Introduction. Coulomb friction or “dry” friction is one of the most basic mechanical phenomena, discussed in virtually every textbook on mechanics. Yet, problems concerning forces, displacements, and evolutions of deformable and rigid bodies subject to friction and unilateral contact are rich with unsettled questions. In rigid body mechanics it has been known since Painlevé in 1895 (and suspected earlier; see Jellet [26]) that there are simple examples where nonuniqueness and nonexistence of continuous solutions appear without impact being present. In recent years these questions have become important in robotics research, and a modern treatment using the theories of complementarity [12] and quasi-variational inequalities [8] was given in [50, 54]. The approach of Moreau [47] and Monteiro Marques [46], where evolutions of rigid impacting bodies are governed by measure differential inclusions, is also an important step forward.

If the assumption of rigidity is removed, there are two levels of mathematical modeling of the physical problem: the first is the discrete level, where the displacement field is represented by a finite number of variables, and the second is the continuous level, where the displacement field is supposed to belong to some infinite-dimensional function space. The present work deals with deformable bodies at the discrete level. Many civil engineering structures like bridges and towers have a discrete representation of the displacements. However, systematic treatments of the discretization of continuous problems to produce discrete representations are provided by methods such as the finite element method and the boundary element method. A brief description of such a process and the peculiarities related to contact problems is given in the appendix. For a more complete presentation we refer to Kikuchi and Oden [29] or Hlaváček et al. [21]. Similarly to the rigid body level and in contrast to the

*Received by the editors January 22, 1996; accepted for publication (in revised form) January 22, 1997.

<http://www.siam.org/journals/siopt/8-2/29784.html>

[†]Department of Mechanical Engineering, Linköping University, S-581 83 Linköping, Sweden (andkl@ikp.liu.se). The research of this author was based on work supported by the Center for Industrial Information Technology (CENIIT) at Linköping University.

[‡]Department of Mathematical Sciences, The Johns Hopkins University, Baltimore, MD 21218-2689 (jsp@vics1.mts.jhu.edu). The research of this author was based on work supported by the National Science Foundation under grant CCR-9213739 and the Office of Naval Research under grant N00014-93-1-0228.

continuous level, at the discrete level it is possible to apply powerful results from complementarity theory to obtain results on existence and multiplicity of solutions and also to construct provably convergent algorithms for numerical solutions. In this paper we state and prove new existence theorems for discrete problems based on ideas extending those developed in [12, 8, 50, 54].

As indicated above the present paper deals with discrete, deformable body problems. To further specify the model, small displacement linear elasticity is used, which means physically that we think of problems where the strains are less than 0.1–0.2% and no substantial superposed rigid body rotations take place. (The theory allows for large superposed rigid body translations.) Furthermore, quasi-static problems are dealt with, meaning that inertia effects are neglected, so physically we model processes which are “slow,” i.e., we think of the evolution as going through a sequence of equilibrium states. Such an assumption is frequently quite valid in, for instance, problems of wear of machine elements such as shrink-fitted shaft and bushing assemblages. It should be noted that even when inertia is neglected, the problem is load path dependent due to the presence of friction. Therefore, it is generally impossible to treat the problem as a static one where we solve for the displacements given the external load: the final displacement state will depend on the history of the load. However, it is possible to introduce a time discretization and convert the problem into a sequence of static-like problems, each one with data depending on the solution of the previous problem in the sequence. It is such an incremental problem that we treat in this paper. The whole modeling process from continuous quasi-static problem to discrete incremental problem is described in the appendix.

The first general formulation of a continuous unilateral contact problem involving friction for a linear elastic body was given in the monograph of Duvaut and Lions [16]. This formulation is for a static friction problem in which Coulomb’s friction law is modified by replacing velocities with displacements and inertia is ignored. Strictly speaking, as already noted, using displacements instead of velocities gives a formulation which has a good physical meaning only when considered as a step in a sequence of incremental problems, i.e., as a problem resulting from a time discretization of the velocity form of Coulomb’s friction law. Essentially due to difficulties with the functional setting, Duvaut and Lions were not able to obtain any results on existence and uniqueness for their problem. The first treatment, not oriented towards numerical algorithms, of the discrete counterpart of the static problem of Duvaut and Lions is due to Janovský [22, 23]. He obtained existence of a solution for all friction coefficients and uniqueness for “small” friction coefficients. A sufficient condition characterizing small friction coefficients in terms of eigenvalues of the positive definite stiffness matrix was given, as well as an abstract example in which nonuniqueness of solutions is present. A sufficient and necessary condition for the existence of a unique solution was given by Alart [1]. For the case of two-dimensional contact or a piecewise linearization of Coulomb’s friction cone, such a condition can also be obtained by means of linear complementarity (LCP) theory; see Klarbring [30]. In Alart et al. [2] and Klarbring [30] the sufficient and necessary condition was numerically investigated for concrete two-dimensional problems.

In contrast to previous works on the discrete incremental friction problem, we investigate in this paper the case where the stiffness matrix is positive semidefinite; i.e., the so-called semicoercive case is treated. Our main results are two new existence theorems for this case. The problem is viewed as an incremental instance in a sequence of problems, which means that an extra load term appears as data as compared to

the original problem of Duvaut and Lions. Physically, the positive semidefiniteness corresponds to a situation where the structure or body is not anchored so that it is free to perform rigid body displacements on top of local deformations. It is frequently quite useful to model engineering contact problems in this way, and it should be noted that in contrast to the bilateral contact case it is not possible to construct “equivalent” problems with nonsingular stiffness matrices by fixing nodes to prevent rigid body displacements. From a mechanical point of view, the fact that the body is not anchored means that certain directions for the external loads are not permissible. Loosely speaking, the external loads have to press the body towards the obstacle. Now, it is intuitively quite clear that the set of external loads that are permissible in this sense will depend on the value of the coefficient of friction: for large such coefficients it is possible to have a larger “tangential” component of the load than for small ones. The first of the two main existence results, Theorem 4.1, is for a large class of friction laws, including nonlinear ones such as the classical and anisotropic versions of Coulomb’s law as well as laws where the friction cone is piecewise linear. The second main existence result, Theorem 4.4, is valid for piecewise linear laws. It takes into account the effect of the value of the friction coefficients and allows for tangential external loads. This result is illustrated by means of a simple 2 degree-of-freedom example.

Following this introduction, section 2 states the governing equations of the discrete incremental version of the problem. For a motivation of these equations starting from the small displacement continuous quasi-static problem, we refer to the appendix. The problem is reformulated as a generalized quasi-variational inequality (GQVI). In fact, the GQVI is a member of a family of problems parametrized by a regularization parameter ε . For $\varepsilon > 0$, members of this family are quasi-variational inequalities (QVI) and $\varepsilon = 0$ corresponds to the original problem. Section 3 contains a proof of existence of solutions to the QVIs for $\varepsilon > 0$. In the first part of section 4, a limiting argument is used to show that the GQVI for $\varepsilon = 0$, and thus the original problem has a solution under conditions on the external loads. It turns out that this first existence result is not broad enough to include a class of important examples. In the second part of section 4, the existence result is therefore strengthened for the case of a polyhedral friction law.

Related work. In this subsection we give a short account of research papers dealing with existence and uniqueness issues of small displacement friction problems. Although related to the present work, these papers do not focus on the discrete incremental problem. For a more general review of frictional contact problems for deformable bodies we refer to the monographs by Kikuchi and Oden [29] and Kalker [27] and the recent proceedings edited by Raous, Jean, and Moreau [52].

As noted above, Duvaut and Lions [16] did not present any results on existence or uniqueness for their continuous static problem. The existence issue for “small” friction coefficients was later studied by Nečas, Jarušek, and Haslinger [48], Jarušek [24, 25], Gastaldi [18], and Kato [28]. The semicoercive case was treated in [25, 18, 28]. Results similar to, but somewhat more restrictive than, Corollary 4.2 of the present paper were given. We should point out that although it may appear that results in the discrete case can easily be obtained from results in the continuous case by simply substituting $\mathbb{R}^n$ for function spaces, this is not entirely true. For instance, trace mappings in the continuous case are onto, but similar matrix objects in the discrete theory do not necessarily have this property and it is not assumed so throughout the present paper.

Results on existence for all friction coefficients and results on uniqueness for small friction coefficients were obtained by using different kinds of regularizations of the problem in Duvaut [15], Demkowicz and Oden [13], Cocu [11], Klarbring, Mikelić, and Shillor [38, 39], Gastaldi [18], Bonfanti [7], and Vy and Ang [55]. The quasi-static problem was treated by Andersson [5, 6] and Klarbring, Mikelić, and Shillor [40]. A simpler version of the quasi-static problem, called the rate problem (see below), was considered in Klarbring [33]. The full dynamic problem, with a normal compliance regularization, was treated by Martins and Oden [44]. Existence of a solution in the semicoercive case with nonlocal friction law was shown in [11]. Again, the conditions are more restrictive than those of the present paper.

If one attempts to study the quasi-static problem of frictional contact by a simplified formulation, an alternative to the incremental problem is the rate problem; i.e., when the solution is known in a time interval $[0, t_0]$, we may pose the problem of determining the rates or right-hand time derivatives of the variables (assuming that they exist) for a given rate of change of the external loading at t_0 . For the discrete case with a linearized friction cone this problem was investigated in Klarbring [32]. It was found that in contrast to the static incremental problem the rate problem may lack a solution for “large” friction coefficients. The uniqueness question on the other hand has the same character for the rate problem as for the incremental problem; i.e., nonuniqueness may be present for large friction coefficients, and the corresponding sufficient and necessary conditions for uniqueness are almost identical. A sufficient condition for uniqueness, very similar to the condition related to the incremental problem by Janovský [22, 23], was given in Doudoumis, Mitsopoulou, and Charalambakis [14]. A two-degree-of-freedom problem investigated in Klarbring [32] was more fully developed in [34] and subsequently used as the starting point for a deep investigation of quasi-static evolutions of such two-degree-of-freedom friction problems in Martins, Monteiro Marques, and Gastaldi [43], Martins et al. [45], and Gastaldi, Monteiro Marques, and Martins [19, 20].

The present paper does not deal with algorithms for solving the discrete deformable body friction problem. A comprehensive algorithmic study of this problem, including the discussion of new algorithms and numerical results, are presented in [9, 10]. See also the review article by Klarbring [36] which discusses the algorithmic aspect of discrete friction problems as it relates to general mathematical programming. In this regard, we should mention the work of Al-Fahed, Stavrolakis, and Panagiotopoulos [4], Al-Fahed and Panagiotopoulos [3], and Panagiotopoulos and Al-Fahed [49], where a static rigid body problem with friction is studied. By treating all obstacles as flexible and using a linearized friction cone, these authors derived an LCP formulation for their problem and applied Fichera’s theory for monotone variational inequalities [17]. They discussed the existence of a solution to the resulting problem and developed solution methods based on Lemke’s algorithm [12].

Frictional contact problems in three dimensions lead to mixed nonlinear complementarity problems; thus they cannot be treated by LCP theory in general. We refer the reader to [50, 54] for a detailed treatment of the solution existence theory for the rigid body contact problem. The present work follows the approach in these references and extends the results in [50] to the deformable body contact problem. The extension is summarized as follows. In [50], under an all-rolling assumption, an existence theorem is established for the rigid body problem. Mathematically speaking, the all-rolling problem is a special case of the deformable body problem with C_n being the negative identity matrix. Furthermore, the improved existence results in this paper,

Theorem 4.4 and Corollaries 4.5 and 4.6, can be applied to the quasi-static rigid body problem [51] for which no existence result is currently available. Mathematically, the latter problem can be considered a special case of the deformable problem with the stiffness matrix K being the zero matrix.

2. Problem statement. In this section, we present the mathematical model for the discrete, three-dimensional, deformable body, small displacement, frictional contact problem and give an equivalent formulation of this problem in terms of a GQVI. The reader is referred to the appendix for the description of the underlying mechanical problem and its continuous-time formulation whose (space,time)-discretization leads to the discrete model presented below.

We explain the notation used in the remainder of this paper. The superscript “ $'$ ” denotes the transpose of vectors and matrices; $u \perp v$ means that the two vectors u and v are perpendicular; $u \circ v$ denotes the Hadamard product of the two vectors u and v (i.e., the i th component of $u \circ v$ is equal to the product $u_i v_i$ for all i); and argmax_x (argmin_x) denotes the optimal solution set of a maximization (minimization, respectively) problem with x as the variable. For a vector x , the support of x , denoted $\operatorname{supp}(x)$, is the index set of the nonzero components of x . For a positive scalar $r > 0$, $\mathbb{B}(0, r)$ denotes the ball in a Euclidean space of appropriate dimension with center at the origin and radius r .

The mathematical model. The data and variables of the model are as follows:

Dimensions.

- n_u = number of displacement nodes;
- n_c = number of contact points;
- $N \equiv 2n_c$, and $N' \equiv n_u + n_c$.

Model data.

- K = an elastic stiffness matrix of order $n_u \times n_u$ and symmetric positive semidefinite; in this paper, we are particularly interested in the case where K is singular;
- C_n, C_t, C_o = static force transformation matrices, each of order $n_c \times n_u$; the subscripts “n, t, o” refer, respectively, to the normal (n) and two tangential components (t and o); the rows of these matrices are denoted C_{in}, C_{it} , and C_{io} , respectively;
- $\mathcal{F}_i : \mathbb{R}_+ \rightarrow \mathbb{R}^2, i = 1, \dots, n_c$ = Coulomb friction law at i th contact; examples of $\mathcal{F}_i(\tau)$ for a given $\tau \geq 0$ include the following [31, 37, 54]:

$$\text{elliptic law} \quad \mathcal{F}_i(\tau) \equiv \{(t_1, t_2) \in \mathbb{R}^2 : \frac{t_1^2}{e_{i1}^2} + \frac{t_2^2}{e_{i2}^2} \leq \tau^2\},$$

$$\text{pyramid law} \quad \mathcal{F}_i(\tau) \equiv \{(t_1, t_2) \in \mathbb{R}^2 : \max(|t_1|, |t_2|) \leq \tau\},$$

$$\text{polygonal law} \quad \mathcal{F}_i(\tau) \equiv \{(t_1, t_2) \in \mathbb{R}^2 : t_1 \cos \theta_{ij} + t_2 \sin \theta_{ij} \leq \tau, \\ j = 1, \dots, r_i\},$$

$$\text{planar law} \quad \mathcal{F}_i(\tau) \equiv \{(t_1, 0) \in \mathbb{R}^2 : |t_1| \leq \tau\},$$

where e_{i1} and e_{i2} are positive constants, θ_{ij} 's are given scalars, and r_i 's are positive integers; the planar law corresponds to a planar problem; note that in the elliptic law, the pyramid law, and the planar law, $\mathcal{F}_i(0) = \{(0, 0)\}$ and $\mathcal{F}_i(\tau)$ contains the origin for $\tau > 0$; the polygonal law, being an approximation of the elliptic law, is required to satisfy these two properties;

- μ = nonnegative n_c -dimensional vector of friction coefficients;
- f_{ext} = n_u -dimensional vector of external forces;
- g = n_c -dimensional constant vector of gaps between contact points and obstacles;
- h_t, h_o = two n_c -dimensional constant vectors in maximum work principle (arising for instance from a time-discretization process of a continuum model).

Model variables.

- u = n_u -dimensional vector of displacements;
- p_n, p_t, p_o = n_c -dimensional vectors of normal and two tangential forces; the components of these vectors are denoted, respectively, p_{in}, p_{it}, p_{io} for $i = 1, \dots, n_c$;
- v_n = n_c -dimensional vector of slacks in obstacle constraints; the components of this vector are denoted v_{in} for $i = 1, \dots, n_c$.

The defining equations of the contact problem are as follows:

Force equilibrium.

$$(1) \quad Ku + C'_n p_n + C'_t p_t + C'_o p_o - f_{\text{ext}} = 0, \quad p_n \geq 0.$$

(Polyhedral) obstacle surface.

$$(2) \quad v_n + C_n u - g = 0, \quad v_n \geq 0;$$

throughout this paper, we assume that the obstacle set

$$\Theta \equiv \{u \in \mathbb{R}^{n_u} : C_n u \leq g\}$$

is nonempty.

Coulomb friction.

$$(3) \quad (p_{it}, p_{io}) \in \mathcal{F}_i(\mu_i p_{in}), \quad i = 1, \dots, n_c.$$

Contact relation.

$$(4) \quad v_n \perp p_n$$

(we recall the nonnegativity of the two vectors v_n and p_n).

Maximum work principle. For $i = 1, \dots, n_c$, (p_{it}, p_{io}) is an optimal solution of the following maximization problem for the variables (q_{it}, q_{io}) :

$$(5) \quad \begin{aligned} & \text{maximize} \quad q_{it}(C_{it}u + h_{it}) + q_{io}(C_{io}u + h_{io}) \\ & \text{subject to} \quad (q_{it}, q_{io}) \in \mathcal{F}_i(\mu_i p_{in}). \end{aligned}$$

This principle is often stated in several equivalent forms, depending on the explicit representation of $\mathcal{F}_i$. For instance, suppose $\mathcal{F}_i(\tau)$ represents the elliptic law. If $p_{in} > 0$, then the pair (p_{it}, p_{io}) satisfies (5) if and only if there exists a scalar λ_i such that

$$h_{it} + C_{it}u = \frac{\lambda_i}{e_{i1}^2} p_{it}, \quad h_{io} + C_{io}u = \frac{\lambda_i}{e_{i2}^2} p_{io},$$

$$0 \geq \left(\frac{p_{it}^2}{e_{i1}^2} + \frac{p_{io}^2}{e_{i2}^2} - \mu_i^2 p_{in}^2 \right) \perp \lambda_i \geq 0,$$

or by solving for λ_i explicitly if and only if

$$\frac{p_{it}^2}{e_{i1}^2} + \frac{p_{io}^2}{e_{i2}^2} \leq \mu_i^2 p_{in}^2,$$

and

$$(6) \quad \begin{aligned} e_{i1}^2 \mu_i p_{in} (h_{it} + C_{it}u) &= \sqrt{e_{i1}^2 (h_{it} + C_{it}u)^2 + e_{i2}^2 (h_{io} + C_{io}u)^2} p_{it}, \\ e_{i2}^2 \mu_i p_{in} (h_{io} + C_{io}u) &= \sqrt{e_{i1}^2 (h_{it} + C_{it}u)^2 + e_{i2}^2 (h_{io} + C_{io}u)^2} p_{io}. \end{aligned}$$

Clearly the last two equations are also valid if $p_{in} = 0$ because in this case both p_{it} and p_{io} must be zero. Similar expressions can be written if $\mathcal{F}_i(\tau)$ is a polygonal approximation of the elliptic law. In particular, for the pyramid law, we see that (p_{it}, p_{io}) satisfies the maximum work principle if and only if there exist slack variables $(s_{it}^\pm, s_{io}^\pm)$ and multipliers $(\lambda_{it}^\pm, \lambda_{io}^\pm)$ such that

$$\begin{aligned} p_{it} + s_{it}^+ &= \mu_i p_{in}, & p_{it} - s_{it}^- &= -\mu_i p_{in}, \\ p_{io} + s_{io}^+ &= \mu_i p_{in}, & p_{io} - s_{io}^- &= -\mu_i p_{in}, \\ h_{it} + C_{it}u &= \lambda_{it}^+ - \lambda_{it}^-, \\ h_{io} + C_{io}u &= \lambda_{io}^+ - \lambda_{io}^-, \\ (s_{it}^\pm, s_{io}^\pm, \lambda_{it}^\pm, \lambda_{io}^\pm) &\geq 0, \\ s_{it}^+ \lambda_{it}^+ &= s_{it}^- \lambda_{it}^- = s_{io}^+ \lambda_{io}^+ = s_{io}^- \lambda_{io}^- = 0. \end{aligned}$$

The latter equivalent formulation of the maximum work principle will be used freely in the proof of Theorem 4.4.

Let $\mathcal{F} : \mathbb{R}_+^{n_c} \rightarrow \mathbb{R}^N$ be the system friction (set-valued) map defined by

$$\mathcal{F}(p_n) \equiv \prod_{i=1}^{n_c} \mathcal{F}_i(\mu_i p_{in}) \quad \text{for } p_n \in \mathbb{R}_+^{n_c}.$$

In terms of this map, the maximum work principle can be stated equivalently in the aggregate form

$$(p_t, p_o) \in \operatorname{argmax} \{ q'_t(h_t + C_t u) + q'_o(h_o + C_o u) : (q_t, q_o) \in \mathcal{F}(p_n) \}.$$

Formulation as a GQVI. In what follows, we reformulate the conditions (1)–(5) as a generalized quasi-variational inequality.

Let $\mathcal{K}$ and F be two set-valued maps from the Euclidean space $\mathbb{R}^n$ into subsets of $\mathbb{R}^n$. Introduced in [8], the GQVI defined by this pair of mappings $(\mathcal{K}, F)$ is to find a vector $x \in \mathcal{K}(x)$ such that

$$\exists y \in F(x) \text{ satisfying } (z - x)'y \geq 0 \quad \forall z \in \mathcal{K}(x).$$

We note that with $x \in \mathcal{K}(x)$, the last inequality holds if and only if

$$x \in \operatorname{argmin} \{ z'y : z \in \mathcal{K}(x) \}.$$

When F is single valued, the generalized quasi-variational inequality $(\mathcal{K}, F)$ becomes the quasi-variational inequality which we will denote using the notation QVI $(\mathcal{K}, F)$.

In order to cast the contact problem as a GQVI, we need to identify the two mappings $\mathcal{K}$ and F . For this purpose, define, for each scalar $\varepsilon \geq 0$, the matrix

$$M(\varepsilon) \equiv \begin{bmatrix} K & C'_n \\ -C_n & 0 \end{bmatrix} + \varepsilon \begin{bmatrix} \mathcal{L} & 0 \\ 0 & \mathcal{M}_n \end{bmatrix} \in \mathbb{R}^{N' \times N'},$$

where

$$\mathcal{L} \equiv \text{diag}(\ell_i) \in \mathbb{R}^{n_u \times n_u}, \ell_i > 0, \text{ for all } i = 1, \dots, n_u$$

and

$$\mathcal{M}_n \equiv \text{diag}(m_{in}) \in \mathbb{R}^{n_c \times n_c}, m_{in} > 0, \text{ for all } i = 1, \dots, n_c$$

are positive diagonal matrices whose diagonal entries are arbitrary and have the physical dimensions of stiffness and flexibility, respectively. From a purely mathematical viewpoint we may think of these additional matrices as identity matrices. The matrix $M(\varepsilon)$ is positive semidefinite for $\varepsilon = 0$ and positive definite for $\varepsilon > 0$. For each pair of vectors $x \equiv (p_t, p_o) \in \mathbb{R}^N$, consider the following mixed LCP in the variables $(u, p_n) \in \mathbb{R}^{N'}$:

$$(7) \quad M(\varepsilon) \begin{pmatrix} u \\ p_n \end{pmatrix} + \begin{pmatrix} C'_t p_t + C'_o p_o - f_{\text{ext}} \\ g \end{pmatrix} = \begin{pmatrix} 0 \\ v_n \end{pmatrix}$$

$$u \text{ free}, \quad 0 \leq p_n \perp v_n \geq 0.$$

For each fixed but arbitrary $\varepsilon > 0$, this mixed LCP has a unique solution which we denote $(u^\varepsilon(x), p_n^\varepsilon(x))$. For $\varepsilon = 0$, this mixed LCP has 0, 1, or ∞ solutions. Let $S^\varepsilon(x) \subseteq \mathbb{R}^{N'}$ denote the solution set of (7) corresponding to $\varepsilon \geq 0$. It is known from linear complementarity theory [12] that $S^0(x)$ is a convex polyhedron, if nonempty. Moreover, for any $(\bar{u}, \bar{p}_n) \in S^0(x)$, $\bar{u}$ is an optimal solution of the convex quadratic program in the variable $u \in \mathbb{R}^{n_c}$:

$$(8) \quad \begin{aligned} & \text{minimize} \quad \frac{1}{2} u' K u + u' (C'_t p_t + C'_o p_o - f_{\text{ext}}) \\ & \text{subject to} \quad u \in \Theta, \end{aligned}$$

and $\bar{p}$ is an optimal multiplier associated with the solution $\bar{u}$. Note that these properties do not apply to the pair $(u^\varepsilon(x), p_n^\varepsilon(x))$ for $\varepsilon > 0$.

For any $\varepsilon \geq 0$, let $F^\varepsilon(x)$ be the collection of vectors of the form

$$-\begin{pmatrix} h_t + C_t u \\ h_o + C_o u \end{pmatrix} + \varepsilon \begin{bmatrix} \mathcal{M}_t & 0 \\ 0 & \mathcal{M}_o \end{bmatrix} x \in \mathbb{R}^N,$$

where $u \in \mathbb{R}^{n_u}$ is such that for some $p_n \in \mathbb{R}^{n_c}$, $(u, p_n) \in S^\varepsilon(x)$. Here $\mathcal{M}_t$ and $\mathcal{M}_o$ are positive diagonal flexibility matrices defined similarly to $\mathcal{M}_n$. Clearly the map $F^\varepsilon : \mathbb{R}^N \rightarrow \mathbb{R}^N$ is single valued for $\varepsilon > 0$ and multivalued for $\varepsilon = 0$. For $x \equiv (p_t, p_o) \in \mathbb{R}^N$ and $\varepsilon \geq 0$, let

$$\mathcal{K}^\varepsilon(x) \equiv \bigcup_{p_n} \mathcal{F}(p_n) \subseteq \mathbb{R}^N,$$

where the union ranges over all elements $(u, p_n) \in S^\varepsilon(x)$. Note that even in the case where $S^\varepsilon(x)$ is a singleton, $\mathcal{K}^\varepsilon(x)$ is not a singleton.

The following result gives the main connection between the contact problem and the GQVI defined by the pair of mappings $(\mathcal{K}^0, F^0)$.

THEOREM 2.1. *If (p_t, p_o) solves the GQVI $(\mathcal{K}^0, F^0)$, then for any $(u, p_n) \in S^0(p_t, p_o)$ the tuple (u, p_n, p_t, p_o) solves the contact problem defined by equations (1)–(5). Conversely, if (u, p_n, p_t, p_o) solves the contact problem defined by the latter equations, then $(u, p_n) \in S^0(p_t, p_o)$; if in addition p_n is the unique vector p'_n such that $(u, p'_n) \in S^0(p_t, p_o)$, then (p_t, p_o) solves the GQVI $(\mathcal{K}^0, F^0)$.*

Proof. For simplicity, we drop the superscript “0” from the mappings $\mathcal{K}$, F , and S . We also write $x \equiv (p_t, p_o)$ throughout.

If x solves the GQVI $(\mathcal{K}, F)$ and $(u, p_n) \in S(x)$, then clearly (u, p_n, p_t, p_o) satisfies (1)–(4). Moreover for all $(q_t, q_o) \in \mathcal{K}(x)$, we have

$$-(q_t - p_t)'(h_t + C_t u) - (q_o - p_o)'(h_o + C_o u) \geq 0.$$

Since $\mathcal{F}(p_n) \subseteq \mathcal{K}(x)$, it follows that the maximum work principle is also satisfied.

For the converse, it suffices to prove the last assertion. For this, it suffices, in turn, to show that for any $(\hat{u}, \hat{p}_n) \in S(x)$, we have $\hat{p}_n = p_n$. (Note that we do not claim that $\hat{u}$ is equal to u .) The proof of this uniqueness result for p_n makes use of a basic property of a convex program proved in Mangasarian [42] which states that for a differentiable convex program, the gradient of the objective function is a constant on the set of optimal solutions of this program. Let $(\hat{u}, \hat{p}_n) \in S(x)$ be given. By the convexity of $S(x)$, it follows that for any scalar $\tau \in (0, 1)$, the pair

$$(u^\tau, p_n^\tau) \equiv \tau(u, p_n) + (1 - \tau)(\hat{u}, \hat{p}_n)$$

remains in $S(x)$; thus u^τ is an optimal solution of the convex quadratic program (8) and p_n^τ is an associated optimal multiplier. By the aforementioned result of Mangasarian, we have $Ku = Ku^\tau$ for all $\tau \in [0, 1]$. Since

$$Ku^\tau + C'_n p_n^\tau + C'_t p_t + C'_o p_o = f_{\text{ext}} \quad \text{for all } \tau \in [0, 1],$$

we deduce

$$0 = C'_n(p_n^\tau - p_n).$$

Restricting τ to be sufficiently close to 1, we have

$$(C_n u < g)_i \Rightarrow (C_n u^\tau < g)_i.$$

By complementarity, it follows that for an index i such that $(C_n u < g)_i$, $p_{in}^\tau = 0$ for all τ sufficiently close to 1. Consequently, for $\tau < 1$ but sufficiently close to 1, we have

$$0 = \sum_{i: p_{in} = (C_n u - g)_i = 0} (C_{in})' p_{in}^\tau + \sum_{i: p_{in} > 0} (C_{in})' (p_{in}^\tau - p_{in}).$$

By assumption, p_n is the unique multiplier corresponding to the optimal solution u . By a result of Kyparisis [41], this implies that the vectors

$$\{(C_{in})' : i \text{ such that } p_{in} > 0\}$$

are linearly independent and there exists a vector w satisfying

$$\begin{aligned} C_{in}w &= 0 \quad \forall i \text{ such that } p_{in} > 0, \\ C_{in}w &< 0 \quad \forall i \text{ such that } p_{in} = C_{in}u - g_i = 0. \end{aligned}$$

Hence we must have $p_{in}^\tau = 0$ for all i such that $p_{in} = C_{in}u - g_i = 0$ and $p_{in}^\tau = p_{in}$ for all i such that $\bar{p}_{in} > 0$. Hence $p_n^\tau = p_n$ for all τ sufficiently close to 1. This implies that $p_n = \hat{p}_n$ as desired. $\square$

It is natural for the reader to ask why we have introduced the maps $\mathcal{K}^\varepsilon$ and F^ε for $\varepsilon > 0$. Actually, in the next section, we shall first show that under appropriate conditions on the friction map $\mathcal{F}$, the QVI $(\mathcal{K}^\varepsilon, F^\varepsilon)$ has a solution for all $\varepsilon > 0$. Subsequently, a limiting argument will be used to show that the GQVI $(\mathcal{K}^0, F^0)$, and thus the contact problem, has a solution under further assumptions on the model data.

3. The QVIs for $\varepsilon > 0$. In this section, we study the QVIs $(\mathcal{K}^\varepsilon, F^\varepsilon)$ for a fixed but arbitrary $\varepsilon > 0$. The following result identifies two key properties of the single valued map F^ε .

PROPOSITION 3.1. *For $\varepsilon > 0$, the map $F^\varepsilon(x)$ is Lipschitz continuous and strongly monotone in $x \equiv (p_t, p_o) \in \mathbb{R}^N$.*

Proof. For simplicity, we drop the superscript ε from the pair $(u^\varepsilon(x), p_n^\varepsilon(x))$ and the function $F^\varepsilon(x)$.

Since $M(\varepsilon)$ is positive definite, it follows that the pair $(u(x), p_n(x))$ is Lipschitz continuous in x ; thus so is

$$F(x) = - \begin{pmatrix} h_t + C_t u(x) \\ h_o + C_o u(x) \end{pmatrix} + \varepsilon \begin{bmatrix} \mathcal{M}_t & 0 \\ 0 & \mathcal{M}_o \end{bmatrix} x.$$

To show the strong monotonicity of F , it suffices to show that

$$(p_t^1 - p_t^2)^T C_t(u(x^1) - u(x^2)) + (p_o^1 - p_o^2)^T C_o(u(x^1) - u(x^2)) \leq 0$$

for any two vectors $x^i \equiv (p_t^i, p_o^i)$, $i = 1, 2$. This follows rather easily from the mixed LCP systems (7) corresponding to x^1 and x^2 by using the positive definiteness of $M(\varepsilon)$. Indeed subtracting the two equations, for $i = 1, 2$,

$$M(\varepsilon) \begin{pmatrix} u^i \\ p_n^i \end{pmatrix} + \begin{pmatrix} C'_t p_t^i + C'_o p_o^i - f_{\text{ext}} \\ g \end{pmatrix} = \begin{pmatrix} 0 \\ v_n^i \end{pmatrix},$$

premultiplying the resulting equation by

$$\begin{pmatrix} u^1 - u^2 \\ p_n^1 - p_n^2 \end{pmatrix}',$$

and using the relations $0 \leq p_n^i \perp v_n^i \geq 0$, we easily establish the desired claim. $\square$

Motivated by the commonly used friction laws (such as the elliptic law and pyramid law), we postulate a mild assumption on the friction maps $\mathcal{F}_i$ for all $i = 1, \dots, n_c$.

Friction assumption (A1). The graph of $\mathcal{F}_i$, denoted $\text{Gr } \mathcal{F}_i$, is a closed convex cone in $\mathbb{R}_+ \times \mathbb{R}^2$; $\mathcal{F}_i(0) = \{(0, 0)\}$; for all $\tau > 0$, $\mathcal{F}_i(\tau)$ contains the origin; and for all scalars $r > 0$, the cut set-valued friction map

$$(9) \quad \tau \in \mathbb{R}_+ \mapsto \mathcal{F}_i(\tau) \cap \mathbb{B}(0, r) \subset \mathbb{R}^2$$

is continuous on $\mathbb{R}_+$.

It can be verified that for both the elliptic law and pyramid law the associated friction map $\mathcal{F}_i$ is indeed a continuous set-valued map; see [50]. For our purpose, the above abstract assumption is enough to yield the following existence result.

THEOREM 3.2. *Under the friction assumption (A1), the QVI $(\mathcal{K}^\varepsilon, F^\varepsilon)$ has a solution for each $\varepsilon > 0$.*

Proof. As above, we drop the superscript ε from the maps $\mathcal{K}^\varepsilon$ and F^ε .

Since F is continuous and strongly monotone, the existence of a solution to the QVI $(\mathcal{K}, F)$ will follow from Corollary 4.1 in [8], provided that we can verify two things: (i) $0 \in \bigcap_{x \in \mathbb{R}^N} \mathcal{K}(x)$ and (ii) for each scalar $r > 0$, the set-valued map

$$x \in \mathbb{R}^N \mapsto \mathcal{K}(x) \cap \left(\prod_{i=1}^{n_c} \mathbb{B}(0, r) \right) \subset \mathbb{R}^N$$

is continuous, where $\mathbb{B}(0, r) \subset \mathbb{R}^2$. Both of these follow easily from the observation that

$$\mathcal{K}(x) = \prod_{i=1}^{n_c} \mathcal{F}_i(\mu_i p_{in}(x))$$

and the friction assumption. $\square$

4. The main results. We introduce another assumption on the friction map $\mathcal{F}_i$. To motivate this assumption, we consider the equations in (6). From these equations, it is easy to show that

$$p_{it}(h_{it} + C_{it}u) \geq 0 \quad \text{and} \quad p_{io}(h_{io} + C_{io}u) \geq 0.$$

Indeed these two inequalities are also valid if $\mathcal{F}_i$ represents the pyramid law. The following is our second assumption on the friction map $\mathcal{F}_i$ for $i = 1, \dots, n_c$.

Friction assumption (A2). For all $\tau \geq 0$ and all $(a_{it}, a_{io}) \in \mathbb{R}^2$,

$$(p_{it}, p_{io}) \in \operatorname{argmax} \{ q_{it}a_{it} + q_{io}a_{io} : (q_{it}, q_{io}) \in \mathcal{F}_i(\tau) \} \Rightarrow \\ \min(a_{it}p_{it}, a_{io}p_{io}) \geq 0.$$

To show the existence of a solution to the GQVI $(\mathcal{K}^0, F^0)$, we take an arbitrary sequence of positive scalars $\{\varepsilon_k\}$ converging to zero. For each k , we take an arbitrary solution $x^k \equiv (p_t^k, p_o^k)$ of the QVI $(\mathcal{K}^{\varepsilon_k}, F^{\varepsilon_k})$ and let $(u^k, p_n^k) \equiv (u^{\varepsilon_k}(x^k), p_n^{\varepsilon_k}(x^k))$. If the sequence $\{(u^k, p_n^k, p_t^k, p_o^k)\}$ is bounded, then every one of its limit points is a desired solution of the contact problem. In the next result, we give a sufficient condition for this sequence to be bounded and verify the claimed solution property of its limit points.

THEOREM 4.1. *Under the friction assumptions (A1) and (A2), if the following implication holds*

$$(10) \quad \left. \begin{aligned} Ku &= C'_n p_n + C'_t p_t + C'_o p_o = 0 \\ 0 &\leq p_n \perp C_n u \leq 0 \\ (p_t, p_o) &\in \operatorname{argmax} \{ q'_t C_t u + q'_o C_o u : (q_t, q_o) \in \mathcal{F}(p_n) \} \end{aligned} \right\} \\ \Rightarrow f'_{ext} u + h'_t p_t + h'_o p_o \leq g' p_n,$$

then the contact problem defined by equations (1)–(5) has a solution.

Proof. For simplicity, we take $\mathcal{L}$ and $\mathcal{M}_\alpha$, for $\alpha = n, t, o$, to be identity matrices of appropriate dimensions.

Let the sequence $\{(u^k, p_n^k, p_t^k, p_o^k)\}$ be as given above. We first show that this sequence is bounded. We may assume by way of contradiction that

$$\lim_{k \rightarrow \infty} \|(u^k, p_n^k, p_t^k, p_o^k)\| = \infty$$

and

$$\lim_{k \rightarrow \infty} \frac{(u^k, p_n^k, p_t^k, p_o^k)}{\|(u^k, p_n^k, p_t^k, p_o^k)\|} = (u^\infty, p_n^\infty, p_t^\infty, p_o^\infty) \neq 0.$$

We claim that the tuple $(u^\infty, p_n^\infty, p_t^\infty, p_o^\infty)$ satisfies the left-hand side of the implication (10). For each k , we have

$$(11) \quad M(\varepsilon_k) \begin{pmatrix} u^k \\ p_n^k \end{pmatrix} + \begin{pmatrix} C'_t p_t^k + C'_o p_o^k - f_{\text{ext}} \\ g \end{pmatrix} = \begin{pmatrix} 0 \\ v_n^k \end{pmatrix},$$

$$u^k \text{ free}, \quad 0 \leq p_n^k \perp v_n^k \geq 0.$$

Normalizing with $\|(u^k, p_n^k, p_t^k, p_o^k)\|$ and passing to the limit $k \rightarrow \infty$, we obtain

$$(12) \quad \begin{aligned} Ku^\infty + C'_n p_n^\infty + C'_t p_t^\infty + C'_o p_o^\infty &= 0, \\ 0 \leq p_n^\infty \perp v_n^\infty \equiv -C_n u^\infty &\geq 0. \end{aligned}$$

Moreover, for each i ,

$$(\mu_i p_{in}^k, p_{it}^k, p_{io}^k) \in \text{Gr } \mathcal{F}_i.$$

Since $\text{Gr } \mathcal{F}_i$ is assumed to be a closed cone, it follows that

$$(\mu_i p_{in}^\infty, p_{it}^\infty, p_{io}^\infty) \in \text{Gr } \mathcal{F}_i;$$

thus $(p_t^\infty, p_o^\infty) \in \mathcal{F}(p_n^\infty)$. Let $(q_t, q_o) \in \mathcal{F}(p_n^\infty)$. For each $i = 1, \dots, n_c$, $(q_{it}, q_{io}) \in \mathcal{F}_i(\mu_i p_{in}^\infty)$. Since $\mathcal{F}_i$ is continuous, thus lower semicontinuous, it follows that there exists a sequence $\{(q_{it}^k, q_{io}^k)\}$ converging to (q_{it}, q_{io}) such that for each k , $(q_{it}^k, q_{io}^k) \in \mathcal{F}_i(\mu_i p_{in}^k / \alpha_k)$, where $\alpha_k \equiv \|(u^k, p_n^k, p_t^k, p_o^k)\|$. Since

$$(13) \quad \begin{pmatrix} p_{it}^k \\ p_{io}^k \end{pmatrix} \in \text{argmin} \left\{ (F^\varepsilon(p_{it}^k, p_{io}^k))' \begin{pmatrix} s_{it} \\ s_{io} \end{pmatrix} : \begin{pmatrix} s_{it} \\ s_{io} \end{pmatrix} \in \mathcal{F}_i(\mu_i p_{in}^k) \right\}$$

and $\alpha_k (q_{it}^k, q_{io}^k) \in \mathcal{F}_i(\mu_i p_{in}^k)$, it follows that

$$(p_{it}^k - \alpha_k q_{it}^k)(C_{it} u^k + h_{it} - \varepsilon_k p_{it}^k) + (p_{io}^k - \alpha_k q_{io}^k)(C_{io} u^k + h_{io} - \varepsilon_k p_{io}^k) \geq 0.$$

Dividing by $\|(u^k, p_n^k, p_t^k, p_o^k)\|^2$ and passing to the limit $k \rightarrow \infty$, we obtain

$$(p_{it}^\infty - q_{it})C_{it} u^\infty + (p_{io}^\infty - q_{io})C_{io} u^\infty \geq 0.$$

Since (q_{it}, q_{io}) is an arbitrary pair in $\mathcal{F}_i(\mu_i p_{in}^\infty)$, it follows that

$$(14) \quad (p_{it}^\infty, p_{io}^\infty) \in \text{argmax} \{ q_{it} C_{it} u^\infty + q_{io} C_{io} u^\infty : (q_{it}, q_{io}) \in \mathcal{F}_i(\mu_i p_{in}^\infty) \}.$$

Since $\mathcal{F}_i(\mu_i p_{in}^\infty)$ contains the origin, we obtain in particular

$$p_{it}^\infty C_{it} u^\infty + p_{io}^\infty C_{io} u^\infty \geq 0 \quad \forall i = 1, \dots, n_c.$$

Premultiplying the first equation in (12) by $(u^\infty)'$, using the complementarity condition in the second equation, and using the above inequalities, we conclude that

$$(u^\infty)' K u^\infty \leq 0.$$

Since K is symmetric positive semidefinite, we deduce that

$$K u^\infty = C_n' p_n^\infty + C_t' p_t^\infty + C_o' p_o^\infty = 0.$$

Consequently, we have established our claim that $(u^\infty, p_n^\infty, p_t^\infty, p_o^\infty)$ satisfies the left-hand side of the implication (10). Thus it follows that

$$(15) \quad f_{\text{ext}}' u^\infty + h_t' p_t^\infty + h_o' p_o^\infty \leq g' p_n^\infty.$$

By (A2) and (13), we deduce that

$$p_{it}^k (C_{it} u^k + h_{it} - \varepsilon_k p_{it}^k) \geq 0 \quad \text{and} \quad p_{io}^k (C_{io} u^k + h_{io} - \varepsilon_k p_{io}^k) \geq 0,$$

which implies

$$(16) \quad p_{it}^k (C_{it} u^k + h_{it}) \geq \varepsilon_k (p_{it}^k)^2 \quad \text{and} \quad p_{io}^k (C_{io} u^k + h_{io}) \geq \varepsilon_k (p_{io}^k)^2.$$

We claim that for all k sufficiently large,

$$(17) \quad p_{it}^k C_{it} u^\infty \geq 0 \quad \text{and} \quad p_{io}^k C_{io} u^\infty \geq 0.$$

We prove this only for the t -component. Clearly the claim is valid if either p_{it}^k or $C_{it} u^\infty$ is zero, so assume that they are both nonzero. For simplicity, assume that $C_{it} u^\infty$ is positive. Since

$$\lim_{k \rightarrow \infty} \frac{C_{it} u^k + h_{it}}{\alpha_k} = C_{it} u^\infty > 0,$$

it follows that for all k sufficiently large $C_{it} u^k + h_{it} > 0$. By (16), it follows that $p_{it}^k > 0$. Thus our claim is established.

By a similar argument, we can establish that for all k sufficiently large,

$$(18) \quad p_{it}^\infty (C_{it} u^k + h_{it}) \geq 0 \quad \text{and} \quad p_{io}^\infty (C_{io} u^k + h_{io}) \geq 0;$$

moreover,

$$p_n^\infty \perp v_n^k \quad \text{and} \quad p_n^k \perp C_n u^\infty.$$

Now, we premultiply (11) by

$$\begin{pmatrix} u^\infty \\ p_n^\infty \end{pmatrix}'.$$

The right-hand product is equal to zero. After taking into account $Ku^\infty = 0$, $(p_n^k)^T C_n u^\infty = 0$, and (17), we deduce that the left-hand product is greater than or equal to

$$\begin{aligned} & \varepsilon_k \left((u^\infty)' u^k + (p_n^\infty)' p_n^k \right) - (p_n^\infty)' C_n u^k + g' p_n^\infty - f'_{\text{ext}} u^\infty \\ &= \varepsilon_k \left((u^\infty)' u^k + (p_n^\infty)' p_n^k \right) + (p_t^\infty)' C_t u^k + (p_o^\infty)' C_o u^k + g' p_n^\infty - f'_{\text{ext}} u^\infty \\ &\geq \varepsilon_k \left((u^\infty)' u^k + (p_n^\infty)' p_n^k \right) + g' p_n^\infty - f'_{\text{ext}} u^\infty - h'_t p_t^\infty - h'_o p_o^\infty \\ &\geq \varepsilon_k \left((u^\infty)' u^k + (p_n^\infty)' p_n^k \right), \end{aligned}$$

where the next to last inequality follows from (18) and the last inequality follows from (15). Consequently, we have

$$0 \geq \varepsilon_k \left((u^\infty)' u^k + (p_n^\infty)' p_n^k \right);$$

this implies that $(u^\infty, p_n^\infty) = 0$. Since $\alpha_k^{-1}(\mu_i p_{in}^k, p_{it}^k, p_{io}^k) \in \text{Gr } \mathcal{F}_i$, by taking limits we deduce that $(p_{it}^\infty, p_{io}^\infty) \in \mathcal{F}_i(0)$. Hence $p_t^\infty = p_o^\infty = 0$.

Summarizing, we conclude that $(u^\infty, p_n^\infty, p_t^\infty, p_o^\infty)$ is equal to zero. This is a contradiction. Hence the sequence $\{(u^k, p_n^k, p_t^k, p_o^k)\}$ must be bounded. Without loss of generality, we may assume that this sequence converges to a tuple $(u^*, p_n^*, p_t^*, p_o^*)$. By using the same limit argument as before for the tuple $(u^\infty, p_n^\infty, p_t^\infty, p_o^\infty)$, we can establish that $(u^*, p_n^*, p_t^*, p_o^*)$ is a solution of the GQVI $(\mathcal{K}, F)$, thus a solution of the contact problem, by Theorem 2.1. $\square$

Several special cases can be obtained from Theorem 4.1. Foremost among these is the frictionless case in which μ is the zero vector. In this case, the contact problem becomes a convex quadratic program and the implication (10) can easily be seen to be necessary and sufficient for this program (and thus the contact problem) to be solvable. The results in the remainder of the paper are most meaningful in the frictional case. The following corollary is important in practice; it requires no proof.

COROLLARY 4.2. *Under the friction assumptions (A1) and (A2), if*

$$\text{Gr } \mathcal{F} \cap (\text{null space of } [C'_n \ C'_t \ C'_o]) = \{(0, 0, 0)\}$$

and

$$(19) \quad \left. \begin{aligned} Ku &= 0 \\ C_n u &\leq 0 \end{aligned} \right\} \Rightarrow f'_{\text{ext}} u \leq 0,$$

then the contact problem defined by equations (1)–(5) has a solution for the tuple $(f_{\text{ext}}, g, h_t, h_o)$, where (g, h_t, h_o) is arbitrary. If instead of (19) the following implication holds

$$(20) \quad \left. \begin{aligned} Ku &= 0 \\ C_n u &\leq 0 \end{aligned} \right\} \Rightarrow u = 0,$$

then the contact problem defined by equations (1)–(5) has a solution for all tuples $(f_{\text{ext}}, g, h_t, h_o)$.

For the interpretation of this corollary note that $Ku = 0$ implies that u is a (small) rigid body displacement. The implication (20) thus says that when the body acts as rigid, the contact constraints, defined by C_n , restrict it from making an unlimited

displacement: the body is trapped by the obstacles. The less restrictive assumption (19) says that when there are directions of escape, i.e., if there exists $u \neq 0$ satisfying $Ku = 0$, $C_n u \leq 0$, then the external force acts in the opposite direction to displacement and prevents the possibility of unlimited displacements. Note also that most finite element schemes will yield matrices C_n , C_t , and C_o which have full row rank. The first condition of Corollary 4.2 is then trivially satisfied.

Polyhedral friction law. Under the first condition of Corollary 4.2, the friction coefficients do not enter the assumptions of this corollary. This indicates that the sufficient condition of Theorem 4.1 may not be broad enough to include some practically important examples. That this is indeed so is shown by the following example, which motivates our next result.

Example. Consider a particle that is placed on a flat rough obstacle surface and make a restriction to the planar case, i.e., there are only the normal and one tangential direction: the “o” component is not present. A cartesian coordinate system, used for reference, is such that the x_1 axis is in the outward normal direction of the obstacle and the x_2 axis is directed tangentially to the obstacle. Although a particle is not a deformable body, such a system may nevertheless be seen as a special case of the mathematical problem considered in this paper; moreover, this system acts as the simplest example of an important class of problems that should be covered by the theory. The following data describes the physical situation: $n_u = 2$, $n_c = 1$, $K = 0$, $C_n = [-1 \ 0]$, $C_t = [0 \ -1]$, and $g = h_t = 0$. If the surface has a friction coefficient that is not zero (it is rough) and if the external force in the x_1 direction is pressing the particle towards the surface, i.e., it is negative, then it is clear that a solution exists for a not too large nonzero tangential force. For instance, if $f_{\text{ext}} = (-1, 1)$, then this problem has a solution if and only if the friction coefficient is greater than or equal to 1. Nevertheless, it is clear that for any nonnegative friction coefficients, the implication (10) fails with $u = (1, 2)$ and $p_n = p_t = 0$. In fact, this implication is necessary and sufficient only for the frictionless case.

The above example calls for a refinement of the condition in Theorem 4.1. To understand what went wrong with the example, let us define the set $\mathcal{S}_\infty$ to consist of all tuples (u, p_n, p_t, p_o) that satisfy the left-hand side of the implication (10). This set is a closed cone in $\mathbb{R}^{N+N'}$. In essence the implication (10) stipulates that the tuple $(f_{\text{ext}}, -g, h_t, h_o)$ belongs to the polar of $\mathcal{S}_\infty$. The refined assumption, to be introduced below, requires that the same tuple belongs to the polar of a proper subset of $\mathcal{S}_\infty$. Since the polar operation reverses the inclusion relation, the new assumption is satisfied by more tuples $(f_{\text{ext}}, g, h_t, h_o)$, and thus we can establish the existence of a solution to more instances of the contact problem.

Define $\mathcal{S}'$ to consist of all tuples $(u, p_n, p_t, p_o) \in \mathcal{S}_\infty$ with the property that the contact problem with the modified data

$$(21) \quad (f_{\text{ext}}, g, h_t, h_o) + (-\mathcal{L}u, \mathcal{M}_n p_n, -\mathcal{M}_t p_t, -\mathcal{M}_o p_o)$$

has solutions of the form

$$(\bar{u}, \bar{p}_n, \bar{p}_t, \bar{p}_o) + \tau(u, p_n, p_t, p_o)$$

for some $(\bar{u}, \bar{p}_n, \bar{p}_t, \bar{p}_o)$ and all $\tau \geq 0$. Using terminology from complementarity theory, we could think of $\mathcal{S}'$ as consisting of all tuples (u, p_n, p_t, p_o) which are “solution rays” of the contact problem with the data (21). From a mechanical point of view $\mathcal{S}'$ consists of rigid body displacement states, which are produced by adding particular

“rigid-body-like” data to the given external data. We illustrate $\mathcal{S}_\infty$ and $\mathcal{S}'$ with the above numerical example.

Example (continued). Since this is a planar problem, we will drop the o -component throughout. It is easy to see that

$$\mathcal{S}_\infty = \{(u_1, u_2, 0, 0) : u_1 \geq 0\}.$$

For an arbitrary element $(u_1, u_2, 0, 0)$ in this set, a general solution of the contact problem with the modified data (21) and $f_{\text{ext}} = (-1, 1)$ is given by

$$(22) \quad (\hat{u}_1, \hat{u}_2) = (0, \hat{u}_2), \quad \hat{p}_n = 1 + \ell_1 u_1 > 0, \quad \hat{p}_t = -1 + \ell_2 u_2,$$

with the following friction conditions satisfied: $|\hat{p}_t| \leq \mu \hat{p}_n$, and with a positive μ ,

$$(23) \quad \begin{aligned} \hat{p}_t &= \mu \hat{p}_n \Rightarrow \hat{u}_2 \leq 0, \\ \hat{p}_t &= -\mu \hat{p}_n \Rightarrow \hat{u}_2 \geq 0, \\ |\hat{p}_t| &< \mu \hat{p}_n \Rightarrow \hat{u}_2 = 0. \end{aligned}$$

With $\mu = 0$, we must have $\hat{p}_t = 0$ and $u_2 = 1/\ell_2$.

Consider the case $\mu > 0$. In order for the tuple $(u_1, u_2, 0, 0)$ to belong to $\mathcal{S}'$, there must exist a tuple $(\bar{u}_1, \bar{u}_2, \bar{p}_n, \bar{p}_t)$ such that for all $\tau \geq 0$, the tuple

$$(\hat{u}_1, \hat{u}_2, \hat{p}_n, \hat{p}_t) \equiv (\bar{u}_1, \bar{u}_2, \bar{p}_n, \bar{p}_t) + \tau (u_1, u_2, 0, 0)$$

satisfies (22) and (23). Since $\hat{u}_1 = 0$, we must have $\bar{u}_1 = u_1 = 0$. Thus $\hat{p}_n = \bar{p}_n = 1$. Moreover, $\hat{p}_t = \bar{p}_t = -1 + \ell_2 u_2$ satisfies $|\bar{p}_t| \leq \mu$. The conditions in (23) become for all $\tau \geq 0$,

$$\begin{aligned} -1 + \ell_2 u_2 &= \mu \Rightarrow \bar{u}_2 + \tau u_2 \leq 0, \\ -1 + \ell_2 u_2 &= -\mu \Rightarrow \bar{u}_2 + \tau u_2 \geq 0, \\ |-1 + \ell_2 u_2| &< \mu \Rightarrow \bar{u}_2 + \tau u_2 = 0. \end{aligned}$$

It follows that

$$u_2 = \begin{cases} 0 & \text{if } \mu \geq 1, \\ \ell_2^{-1}(1 - \mu) & \text{if } \mu \in [0, 1). \end{cases}$$

Consequently,

$$\mathcal{S}' = \begin{cases} \{(0, 0, 0, 0)\} & \text{if } \mu \geq 1, \\ \{(0, \ell_2^{-1}(1 - \mu), 0, 0)\} & \text{if } \mu \in [0, 1). \end{cases}$$

We note that $\mathcal{S}'$ is a much smaller set than $\mathcal{S}_\infty$.

In what follows, we shall deal with the friction pyramid law. Although a similar treatment can be applied to more general polyhedral laws (like a polygonal approximation of the elliptic law), we will omit the details of the generalization. Unfortunately, due to a technical difficulty, our analysis is not applicable to the elliptic law. The main reason is that our proof of the improved existence result, Theorem 4.4 below, is based on a technical lemma which concerns a mixed LCP.

LEMMA 4.3. *Consider the mixed LCP*

$$\begin{aligned} w &= Ax + By + a, \\ 0 &= Cx + Dy + b, \end{aligned}$$

$$0 \leq x \perp w \geq 0, \quad y \text{ free},$$

with $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{m \times n}$, $D \in \mathbb{R}^{m \times m}$, $a \in \mathbb{R}^n$, and $b \in \mathbb{R}^m$. Let $\{(a^k, b^k)\}$ be a sequence of vectors, and for each k let (w^k, x^k, y^k) be a solution of the mixed LCP corresponding to (a^k, b^k) . If $\{(a^k, b^k)\}$ converges to $(\bar{a}, \bar{b})$, then there exists a solution $(\bar{w}, \bar{x}, \bar{y})$ of the mixed LCP corresponding to $(\bar{a}, \bar{b})$ such that for infinitely many k ,

$$(24) \quad \text{supp}(\bar{w}) \times \text{supp}(\bar{x}) \subseteq \text{supp}(w^k) \times \text{supp}(x^k);$$

thus $\bar{w}'x^k = \bar{x}'w^k = 0$.

Proof. The fact that the limiting mixed LCP corresponding to $(\bar{a}, \bar{b})$ has a solution follows from a result of Robinson [53] concerning the locally upper Lipschitzian property of a polyhedral multifunction. Nevertheless, our assertion is stronger than just the existence of a solution to the limiting mixed LCP; we claim that the latter problem has a solution that satisfies the complementarity conditions in (24) for infinitely many k . This additional conclusion is not contained in Robinson's work. Instead, the lemma is similar in spirit to Theorem 1.3.4 in [12], which concerns the standard LCP.

Since there are only finitely many index sets, it follows that there exist two complementary subsets $\mathcal{J}_w$ and $\mathcal{J}_x$ of $\{1, \dots, n\}$ and a subsequence $\{(w^k, x^k, y^k) : k \in \kappa\}$ for some subset κ of $\{1, 2, \dots, \infty\}$ such that for all $k \in \kappa$,

$$\text{supp}(w^k) \times \text{supp}(x^k) = \mathcal{J}_w \times \mathcal{J}_x.$$

Consider the system of linear inequalities

$$\begin{aligned} w &= Ax + By + a, \\ 0 &= Cx + Dy + b, \end{aligned}$$

$$\begin{aligned} (w, x) &\geq 0, \\ w_i &= 0 \quad \text{for all } i \notin \mathcal{J}_w, \\ x_i &= 0 \quad \text{for all } i \notin \mathcal{J}_x. \end{aligned}$$

For each $k \in \kappa$, the triple (w^k, x^k, y^k) is a solution of this system corresponding to (a^k, b^k) . From linear inequality theory, it follows that in the limit there exists a solution $(\bar{w}, \bar{x}, \bar{y})$ to the above system of linear inequalities corresponding to $(\bar{a}, \bar{b})$. This triple $(\bar{w}, \bar{x}, \bar{y})$ satisfies the properties asserted by the lemma. $\square$

With the above lemma, we can now establish our improved result for the contact model with the friction pyramid law.

THEOREM 4.4. *For the friction pyramid law, if*

$$f'_{ext}u + h'_t p_t + h'_o p_o \leq g' p_n$$

for all $(u, p_n, p_t, p_o) \in \mathcal{S}'$, then the contact problem defined by equations (1)–(5) has a solution.

Proof. As in the proof of Theorem 4.1, we take for simplicity $\mathcal{L}$ and $\mathcal{M}_\alpha$, for $\alpha = n, t, o$, to be identity matrices of appropriate dimensions.

By Theorem 3.2, for each $\varepsilon_k > 0$, the QVI $(\mathcal{K}^\varepsilon, F^\varepsilon)$ has a solution (p_t^k, p_o^k) . Adding slacks $(s_{it}^\pm, s_{io}^\pm)$ to the friction pyramid law,

$$\max(|p_{it}|, |p_{io}|) \leq \mu_i p_{in},$$

and writing out the maximum work principle by introducing suitable multipliers $(\lambda_{it}^\pm, \lambda_{io}^\pm)$, we deduce that the following mixed linear complementarity system has a solution $(u^k, p_n^k, p_t^k, p_o^k, \lambda_t^{k\pm}, \lambda_o^{k\pm}, s_t^{k\pm}, s_o^{k\pm})$:

$$\begin{aligned} & \begin{bmatrix} K & C'_n \\ -C_n & 0 \end{bmatrix} \begin{pmatrix} u^k \\ p_n^k \end{pmatrix} + \begin{pmatrix} C'_t p_t^k + C'_o p_o^k - f_{\text{ext}} + \varepsilon_k u^k \\ g + \varepsilon_k p_n^k \end{pmatrix} = \begin{pmatrix} 0 \\ v_n^k \end{pmatrix}, \\ & u^k \text{ free}, \quad 0 \leq p_n^k \perp v_n^k \geq 0, \\ & p_t^k + s_t^{k+} = \mu \circ p_n^k, \quad p_t^k - s_t^{k-} = -\mu \circ p_n^k, \\ & p_o^k + s_o^{k+} = \mu \circ p_n^k, \quad p_o^k - s_o^{k-} = -\mu \circ p_n^k, \\ (25) \quad & h_t + C_t u^k - \varepsilon_k p_t^k = \lambda_t^{k+} - \lambda_t^{k-}, \\ & h_o + C_o u^k - \varepsilon_k p_o^k = \lambda_o^{k+} - \lambda_o^{k-}, \\ & 0 \leq \lambda_t^{k+} \perp s_t^{k+} \geq 0, \\ & 0 \leq \lambda_t^{k-} \perp s_t^{k-} \geq 0, \\ & 0 \leq \lambda_o^{k+} \perp s_o^{k+} \geq 0, \\ & 0 \leq \lambda_o^{k-} \perp s_o^{k-} \geq 0. \end{aligned}$$

Without loss of generality, we may take the multipliers $(\lambda_t^{k\pm}, \lambda_o^{k\pm})$ to satisfy the complementarity conditions

$$(26) \quad \lambda_t^{k+} \perp \lambda_t^{k-} \quad \text{and} \quad \lambda_o^{k+} \perp \lambda_o^{k-}.$$

Either by direct verification or as in the proof of Theorem 4.1, we have

$$(27) \quad p_{it}^k (h_{it} + C_{it} u^k) \geq \varepsilon_k (p_{it}^k)^2 \quad \text{and} \quad p_{io}^k (h_{io} + C_{io} u^k) \geq \varepsilon_k (p_{io}^k)^2.$$

Unlike the proof of the previous theorem, we show instead that the sequence

$$(28) \quad \{\varepsilon_k(u^k, p_n^k, p_t^k, p_o^k)\}$$

is bounded. Indeed premultiplying the first equation in the above complementarity system by

$$\begin{pmatrix} u^k \\ p_n^k \end{pmatrix}',$$

by the positive semidefiniteness of K , and by (27), and by simplifying, we deduce that

$$\varepsilon_k ((u^k)' u^k + (p_n^k)' p_n^k + (p_t^k)' p_t^k + (p_o^k)' p_o^k) \leq f'_{\text{ext}} u^k - g' p_n^k + h'_t p_t^k + h'_o p_o^k.$$

Consequently, we obtain

$$\varepsilon_k \|(u^k, p_n^k, p_t^k, p_o^k)\| \leq \|(f_{\text{ext}}, g, p_t, p_o)\| \quad \text{for all } k,$$

which shows that the sequence (28) is bounded. Thus so is the sequence of (scaled) slacks $\{\varepsilon_k(v_n^k, s_t^{k\pm}, s_o^{k\pm})\}$. Taking into account (26), it follows that the sequence of (scaled) multipliers $\{\varepsilon_k(\lambda_t^{k\pm}, \lambda_o^{k\pm})\}$ is also bounded. Without loss of generality, we may assume that

$$(29) \quad \begin{aligned} \lim_{k \rightarrow \infty} \varepsilon_k (u^k, p_n^k, p_t^k, p_o^k, v_n^k, \lambda_t^{k\pm}, \lambda_o^{k\pm}, s_t^{k\pm}, s_o^{k\pm}) \\ = (u^\infty, p_n^\infty, p_t^\infty, p_o^\infty, v_n^\infty, \lambda_t^{\infty\pm}, \lambda_o^{\infty\pm}, s_t^{\infty\pm}, s_o^{\infty\pm}). \end{aligned}$$

We claim that the tuple $(u^\infty, p_n^\infty, p_t^\infty, p_o^\infty)$ is an element of the set $\mathcal{S}'$. By using an argument similar to the one in Theorem 4.1, we can first establish that $(u^\infty, p_n^\infty, p_t^\infty, p_o^\infty)$ belongs to $\mathcal{S}^\infty$. Indeed the limit tuple (29) must satisfy the following conditions:

$$\begin{aligned} Ku^\infty &= C'_n p_n^\infty + C'_t p_t^\infty + C'_o p_o^\infty = 0, \\ u^\infty \text{ free}, \quad 0 &\leq p_n^\infty \perp v_n = -C_n u^\infty \geq 0, \\ p_t^\infty + s_t^{\infty+} &= \mu \circ p_n^\infty, \quad p_t^\infty - s_t^{\infty-} = -\mu \circ p_n^\infty, \\ p_o^\infty + s_o^{\infty+} &= \mu \circ p_n^\infty, \quad p_o^\infty - s_o^{\infty-} = -\mu \circ p_n^\infty, \\ C_t u^\infty &= \lambda_t^{\infty+} - \lambda_t^{\infty-}, \\ C_o u^\infty &= \lambda_o^{\infty+} - \lambda_o^{\infty-}, \\ 0 &\leq \lambda_t^{\infty+} \perp s_t^{\infty+} \geq 0, \\ 0 &\leq \lambda_t^{\infty-} \perp s_t^{\infty-} \geq 0, \\ 0 &\leq \lambda_o^{\infty+} \perp s_o^{\infty+} \geq 0, \\ 0 &\leq \lambda_o^{\infty-} \perp s_o^{\infty-} \geq 0. \end{aligned}$$

By considering $\varepsilon_k(u^k, p_n^k, p_t^k, p_o^k)$ as part of the constant vector, the system (25) is a mixed LCP in the variables

$$(u^k, p_n^k, p_t^k, p_o^k, v_n^k, \lambda_t^{k\pm}, \lambda_o^{k\pm}, s_t^{k\pm}, s_o^{k\pm}).$$

Thus by Lemma 4.3 there exists

$$(\bar{u}, \bar{p}_n, \bar{p}_t, \bar{p}_o, \bar{v}_n, \bar{\lambda}_t^\pm, \bar{\lambda}_o^\pm, \bar{s}_t^\pm, \bar{s}_o^\pm)$$

satisfying the mixed LCP system

$$\begin{aligned}
(30) \quad & \begin{bmatrix} K & C'_n \\ -C_n & 0 \end{bmatrix} \begin{pmatrix} \bar{u} \\ \bar{p}_n \end{pmatrix} + \begin{pmatrix} C'_t \bar{p}_t + C'_o \bar{p}_o - f_{\text{ext}} + u^\infty \\ g + p_n^\infty \end{pmatrix} = \begin{pmatrix} 0 \\ \bar{v}_n \end{pmatrix}, \\
& \bar{u} \text{ free}, \quad 0 \leq \bar{p}_n \perp \bar{v}_n \geq 0, \\
& \bar{p}_t + \bar{s}_t^+ = \mu \circ \bar{p}_n, \quad \bar{p}_t - \bar{s}_t^- = -\mu \circ \bar{p}_n, \\
& \bar{p}_o + \bar{s}_o^+ = \mu \circ \bar{p}_n, \quad \bar{p}_o - \bar{s}_o^- = -\mu \circ \bar{p}_n, \\
& h_t + C_t \bar{u} - p_t^\infty = \bar{\lambda}_t^+ - \bar{\lambda}_t^-, \\
& h_o + C_o \bar{u} - p_o^\infty = \bar{\lambda}_o^+ - \bar{\lambda}_o^-, \\
& 0 \leq \bar{\lambda}_t^+ \perp \bar{s}_t^+ \geq 0, \\
& 0 \leq \bar{\lambda}_t^- \perp \bar{s}_t^- \geq 0, \\
& 0 \leq \bar{\lambda}_o^+ \perp \bar{s}_o^+ \geq 0, \\
& 0 \leq \bar{\lambda}_o^- \perp \bar{s}_o^- \geq 0;
\end{aligned}$$

moreover, for infinitely many k , we have

$$\begin{aligned}
& \text{supp}(\bar{p}_n) \times \text{supp}(\bar{v}_n) \times \text{supp}(\bar{s}_t^\pm) \times \text{supp}(\bar{s}_o^\pm) \times \text{supp}(\bar{\lambda}_t^\pm) \times \text{supp}(\bar{\lambda}_o^\pm) \subseteq \\
& \text{supp}(p_n^k) \times \text{supp}(v_n^k) \times \text{supp}(s_t^{k\pm}) \times \text{supp}(s_o^{k\pm}) \times \text{supp}(\lambda_t^{k\pm}) \times \text{supp}(\lambda_o^{k\pm}).
\end{aligned}$$

Using this inclusion and the limit (29), we can easily verify the following orthogonality conditions:

$$\begin{aligned}
& \bar{p}_n \perp v_n^\infty, \quad \bar{v}_n \perp p_n^\infty, \\
& \bar{s}_t^\pm \perp \lambda_t^{\infty\pm}, \quad \bar{\lambda}_t^\pm \perp s_t^{\infty\pm}, \\
& \bar{s}_o^\pm \perp \lambda_o^{\infty\pm}, \quad \bar{\lambda}_o^\pm \perp s_o^{\infty\pm}.
\end{aligned}$$

With these conditions, we can show that for all scalars $\tau \geq 0$, the tuple

$$\begin{aligned}
& (\hat{u}, \hat{p}_n, \hat{p}_t, \hat{p}_o, \hat{v}_n, \hat{\lambda}_t^\pm, \hat{\lambda}_o^\pm, \hat{s}_t^\pm, \hat{s}_o^\pm) \equiv (\bar{u}, \bar{p}_n, \bar{p}_t, \bar{p}_o, \bar{v}_n, \bar{\lambda}_t^\pm, \bar{\lambda}_o^\pm, \bar{s}_t^\pm, \bar{s}_o^\pm) + \\
& \tau (u^\infty, p_n^\infty, p_t^\infty, p_o^\infty, v_n^\infty, \lambda_t^{\infty\pm}, \lambda_o^{\infty\pm}, s_t^{\infty\pm}, s_o^{\infty\pm})
\end{aligned}$$

remains a solution of the mixed LCP (30) with the constant vector

$$(f_{\text{ext}}, g, h_t, h_o) + (-u^\infty, p_n^\infty, -p_t^\infty, -p_o^\infty).$$

This establishes the claim that $(u^\infty, p_n^\infty, p_t^\infty, p_o^\infty) \in \mathcal{S}'$. Consequently, by assumption, it follows that

$$f'_{\text{ext}} u^\infty + h'_t p_t^\infty + h'_o p_o^\infty \leq g' p_n^\infty.$$

By the same argument as in the proof of Theorem 4.1, we can now establish that $(u^\infty, p_n^\infty, p_t^\infty, p_o^\infty)$ is the zero tuple. From this, it follows that $(\bar{u}, \bar{p}_n, \bar{p}_t, \bar{p}_o)$ is a solution of the contact problem. $\square$

From the above proof, it is clear why we have restricted Theorem 4.4 to the friction pyramid law (or more generally, a friction polygonal law); namely, this restriction is needed in order to exploit an equivalent mixed LCP formulation of the contact

problem. Although we believe that Theorem 4.4 will remain valid under the elliptic law, we do not have a proof or counterexample for this conjecture.

We next give a special case of Theorem 4.4 that serves to illustrate the assumption of the theorem. This case is of considerable practical importance; in particular, it is applicable to the planar 1-point contact problem discussed previously. As we have mentioned, most finite element discretization schemes will satisfy the assumption on the null space of C'_n , C'_t , and C'_o .

Before stating the result, we give a physical interpretation of the assumptions. Conditions (i) and (ii) below define the rigid body displacements u that are compatible with the constraints, i.e., they identify critical directions in which the body can move to infinity without interfering with the constraints imposed by the obstacles. The element $(\bar{u}, \bar{p}_n, \bar{p}_t, \bar{p}_o)$, referred to in condition (iii), is a state compatible with the rigid body displacement u in the sense of the last two requirements in (31). This critical state is produced by subtracting a force proportional to the displacement, $\mathcal{L}u$, from the given external force. Now the second assumption of the corollary says that for any displacement u satisfying conditions (i), (ii), and (iii), the external force f_{ext} should be in the opposite direction to the direction u of rigid body escape.

COROLLARY 4.5. *Under the friction pyramid law, assume*

$$\text{Gr } \mathcal{F} \cap (\text{null space of } [C'_n \ C'_t \ C'_o]) = \{(0, 0, 0)\}.$$

Assume further that for some positive diagonal matrix $\mathcal{L}$, $f'_{\text{ext}}u \leq 0$ for all u satisfying (i) $Ku = 0$, (ii) $C_n u \leq 0$, and (iii) there exists $(\bar{u}, \bar{p}_n, \bar{p}_t, \bar{p}_o)$ such that

$$(31) \quad \begin{aligned} K\bar{u} + C'_n \bar{p}_n + C'_t \bar{p}_t + C'_o \bar{p}_o &= f_{\text{ext}} - \mathcal{L}u, \\ C_n u \perp \bar{p}_n &\geq 0, \end{aligned}$$

$$(\bar{p}_t, \bar{p}_o) \in \operatorname{argmax} \{q'_t C_t u + q'_o C_o u : (q_t, q_o) \in \mathcal{F}(\bar{p}_n)\}.$$

Then the contact problem defined by equations (1)–(5) has a solution for the tuple $(f_{\text{ext}}, g, h_t, h_o)$, where (g, h_t, h_o) is arbitrary.

Proof. Let (u, p_n, p_t, p_o) be an arbitrary element of $\mathcal{S}'$. We must have $p_n = p_t = p_o = 0$, and u satisfies conditions (i) and (ii). Let $(\bar{u}, \bar{p}_n, \bar{p}_t, \bar{p}_o)$ be such that for all $\tau \geq 0$, $(\bar{u} + \tau u, \bar{p}_n, \bar{p}_t, \bar{p}_o)$ is a solution of the contact problem with the modified data $(f_{\text{ext}} - \mathcal{L}u, g, h_t, h_o)$. Thus we have for $\tau \geq 0$,

$$\bar{p}'_n(g - C_n \bar{u} - \tau C_n u) = 0,$$

and $(\bar{p}_t, \bar{p}_o)$ maximizes

$$q'_t(h_t + C_t \bar{u} + \tau C_t u) + q'_o(h_o + C_o \bar{u} + \tau C_o u)$$

over all $(q_t, q_o) \in \mathcal{F}(\bar{p}_n)$. These two conditions imply that

$$\bar{p}_n \perp C_n u,$$

and $(\bar{p}_t, \bar{p}_o)$ maximizes $q'_t C_t u + q'_o C_o u$ over all $(q_t, q_o) \in \mathcal{F}(\bar{p}_n)$. Consequently, it follows by assumption that

$$0 \geq f'_{\text{ext}}u = f'_{\text{ext}}u + h'_t p_t + h'_o p_o - g' p_n.$$

Hence all the assumptions of Theorem 4.4 are satisfied, and the desired conclusion follows. $\square$

In general, if a vector u satisfies the three conditions (i)–(iii), then premultiplying the first equation in (31) by u' and using the complementarity condition $\bar{p}'C_n u = 0$ and the fact that $\bar{p}'_t C_t u + \bar{p}'_o C_o u \geq 0$ (because $0 \in \mathcal{F}(\bar{p}_n)$), we deduce that

$$f'_{\text{ext}} u \geq u' \mathcal{L} u.$$

Consequently, assuming that $f'_{\text{ext}} u$ is nonpositive for all vectors u satisfying (i)–(iii) is equivalent to saying that the only vector u satisfying conditions (i)–(iii) is the zero vector. Thus we can rephrase Corollary 4.5 in the following way.

COROLLARY 4.6. *Under the friction pyramid law, if*

$$\text{Gr } \mathcal{F} \cap (\text{null space of } [C'_n \ C'_t \ C'_o]) = \{(0, 0, 0)\}$$

and for some positive diagonal matrix $\mathcal{L}$,

$$\left. \begin{aligned} Ku &= 0 \\ K\bar{u} + C'_n \bar{p}_n + C'_t \bar{p}_t + C'_o \bar{p}_o &= f_{\text{ext}} - \mathcal{L} u \\ 0 &\geq C_n u \perp \bar{p}_n \geq 0 \\ (\bar{p}_t, \bar{p}_o) &\in \text{argmax} \{ q'_t C_t u + q'_o C_o u : (q_t, q_o) \in \mathcal{F}(\bar{p}_n) \} \end{aligned} \right\} \Rightarrow u = 0,$$

then the contact problem defined by equations (1)–(5) has a solution for the tuple $(f_{\text{ext}}, g, h_t, h_o)$, where (g, h_t, h_o) is arbitrary.

Loosely speaking, this corollary says that if it is not possible to produce critical states $(\bar{u}, \bar{p}_n, \bar{p}_t, \bar{p}_o)$ compatible with a nonzero rigid body displacement by subtracting a force proportional to the displacement to the external force, then the problem has a solution.

If not for the specific friction law, Corollary 4.6 is easily seen to be a generalization of Corollary 4.2. Whether Corollary 4.6 will remain valid under the broad friction assumptions (A1) and (A2) is an open question at this time.

Example (final analysis). Consider the planar 1-point contact problem with an arbitrary force vector $f_{\text{ext}} = (f_1, f_2)$ and friction coefficient $\mu > 0$. We wish to determine the set of displacements u that satisfy conditions (i), (ii), and (iii) in Corollary 4.5. For such a vector u and a pair $(\bar{p}_n, \bar{p}_t)$ satisfying (31), we have

$$\begin{aligned} \bar{p}_n &= -f_1 + \ell_1 u_1 \geq 0, & u_1 &\geq 0, & \bar{p}_n u_1 &= 0, \\ \bar{p}_t &= -f_2 + \ell_2 u_2, & |\bar{p}_t| &\leq \mu \bar{p}_n, \\ \bar{p}_t &\in \text{argmax} \{-q_t u_2 : |q_t| \leq \mu \bar{p}_n\}. \end{aligned}$$

By an easy calculation, we can deduce that

$$(\ell_1 u_1, \ell_2 u_2) = \begin{cases} (f_1, f_2) & \text{if } f_1 \geq 0, \\ (0, 0) & \text{if } f_1 < 0 \text{ and } |f_2| \leq \mu |f_1|, \\ (0, f_2 - \mu |f_1|) & \text{if } f_1 < 0 \text{ and } f_2 > \mu |f_1|, \\ (0, \mu |f_1| - f_2) & \text{if } f_1 < 0 \text{ and } f_2 < -\mu |f_1|. \end{cases}$$

Consequently, if either $(f_1, f_2) = (0, 0)$ or $[f_1 < 0 \text{ and } |f_2| \leq \mu |f_1|]$, then the assumptions of Corollary 4.6 are satisfied; for these forces, the contact problem thus has a solution. For all other forces, the implication in the corollary is violated; it is not difficult to verify that the contact problem has no solution for these forces.

5. Appendix. In the appendix we give a formulation in terms of vector fields defined over (open) sets of $\mathbb{R}^3$. The aim is to motivate the discrete problem studied in section 2. No attempt is made to give a precise mathematical formulation and analysis of the continuous problem. Rather, we perform a formal finite element discretization in order to derive the discrete problem. There are of course alternative discretization schemes that can be used to obtain the discrete problems. For instance, the boundary element method is a possible candidate which it appears natural to use for some contact problems.

5.1. Continuous-time frictional contact problems. In this section we give a mathematical formulation of the physical situation of a loaded deformable solid body supported by a rough stiff foundation (obstacle) that is considered as rigid. For the deformable solid we use a model of linear elasticity. For the modeling of contact with the foundation we use Signorini's unilateral contact conditions coupled with a generalized Coulomb's friction law. Note that, in the small displacement case we are considering, the situation when two deformable bodies come into contact is mathematically equivalent to the situation with one deformable and one rigid body. Note also that our basic problem is a quasi-static evolution problem, where Coulomb's friction law appears with a time derivative. Thus, the problem is different from the static friction problem posed in Duvaut and Lions [16] and considered in a large number of references. However, after an implicit time discretization our problem will resemble that of Duvaut and Lions; see section 5.2.

The deformable body occupies an open set Ω of $\mathbb{R}^3$. A system of orthonormal axes $0x_1x_2x_3$ is used for reference. The boundary $\partial\Omega$ of Ω consists of disjoint parts S_u , S_t , and S_c and is assumed smooth so that normal and tangential directions of this two-dimensional surface can be defined everywhere. Of special interest will be the case when $S_u = \emptyset$. Using a component notation and the summation convention for indices i, j, k , and l we have the following basic equations of linear elasticity:

$$(32) \quad \left. \begin{aligned} \partial\sigma_{ij}^f/\partial x_j + \bar{f}_i^f &= 0 \\ \sigma_{ij}^f &= E_{ijkl}^f \partial u_k^f/\partial x_l \end{aligned} \right\} \text{ in } \Omega,$$

$$(33) \quad u_i^f = \bar{u}_i^f \text{ on } S_u,$$

$$(34) \quad \sigma_{ij}^f \nu_j^f = \bar{t}_i^f \text{ on } S_t,$$

where the following notations are used:

- σ_{ij}^f = components of the stress tensor;
- E_{ijkl}^f = elasticity constants, which are assumed to have the usual symmetry and ellipticity properties;
- $\bar{f}_i^f$ = prescribed body forces;
- $\bar{t}_i^f$ = prescribed surface tractions;
- u_i^f = components of the displacement vector with prescribed components $\bar{u}_i^f$ on S_u ;
- ν_i^f = components of the outward unit normal vector of $\partial\Omega$.

In general, throughout the paper, a superscript f denotes a field quantity, i.e., a quantity defined over a subset of $\mathbb{R}^3$. The prescribed external force quantities $\bar{f}_i^f$ and $\bar{t}_i^f$ are generally dependent on the time t , making the unknown fields σ_{ij}^f and u_i^f also dependent on time. For simplicity, this dependence is not explicitly expressed

throughout the following discussion; furthermore, we assume that the prescribed displacements on S_u are zero; i.e., $\bar{u}_i^f = 0$.

The boundary part S_c is the potential contact surface where the elastic body may come into contact with the rigid obstacle. To give the conditions governing rough contact, we represent the displacement and the traction on S_c by means of vector fields

$$\begin{pmatrix} w_n^f \\ w_t^f \\ w_o^f \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} p_n^f \\ p_t^f \\ p_o^f \end{pmatrix},$$

respectively. We define, at each point of S_c , a 3-by-3 matrix C^f and a prescribed vector $(g_n^f, g_t^f, g_o^f)'$ of initial distances. The displacement, u_i^f , and the traction, $t_i^f \equiv \sigma_{ij}^f \nu_j^f$, are then related to these new representations of displacement and traction in the following way:

$$(35) \quad \begin{pmatrix} w_n^f \\ w_t^f \\ w_o^f \end{pmatrix} = \begin{pmatrix} g_n^f \\ g_t^f \\ g_o^f \end{pmatrix} - C^f \begin{pmatrix} u_1^f \\ u_2^f \\ u_3^f \end{pmatrix}, \quad \begin{pmatrix} t_1^f \\ t_2^f \\ t_3^f \end{pmatrix} = -(C^f)' \begin{pmatrix} p_n^f \\ p_t^f \\ p_o^f \end{pmatrix}.$$

The first row of C^f , denoted C_n^f , is a unit vector representing the outward unit normal of S_c ; i.e., its components are equal to ν_i^f . The second and third rows of C^f , denoted C_t^f and C_o^f , are unit vectors such that the three rows represent an orthonormal basis of $\mathbb{R}^3$ and, thus, these two rows span the tangent plane of S_c . We are now in a position to state Signorini's contact condition as the following complementarity condition:

$$(36) \quad p_n^f \geq 0, \quad w_n^f \geq 0, \quad p_n^f w_n = 0 \quad \text{on } S_c.$$

This condition governs the behavior of normal components of fields defined on the contact surface. The tangential components are governed by a law of friction, which is a phenomenon which has been given numerous mathematical formulations. We would like to cover a large class of such formulations. To that end, we introduce, at each point of S_c , a general set-valued function $\mathcal{F}^f : \mathbb{R}_+ \rightarrow \mathbb{R}^2$ representing the admissible friction tractions. Explicit examples, previously given in the literature, include the following:

$$\begin{aligned} \text{elliptic law} \quad \mathcal{F}^f(\tau) &\equiv \{(q_t, q_o) \in \mathbb{R}^2 : q_t^2/e_1^2 + q_o^2/e_2^2 \leq \tau^2\}, \\ \text{pyramid law} \quad \mathcal{F}^f(\tau) &\equiv \{(q_t, q_o) \in \mathbb{R}^2 : \max(|q_t|, |q_o|) \leq \tau\}, \\ \text{polygonal law} \quad \mathcal{F}^f(\tau) &\equiv \{(q_t, q_o) \in \mathbb{R}^2 : q_t \cos \theta_j + q_o \sin \theta_j \leq \tau, \\ &\quad j = 1, \dots, r\}, \end{aligned}$$

where e_1 and e_2 are positive constants, θ_j 's are given scalars, and r 's are positive integers. All these parameters should be regarded as fields over S_c . However, for notational simplicity no superscript f is used here. The classical isotropic Coulomb's friction law is obtained from the elliptic law by putting $e_1 = e_2 = 1$. Given $\mathcal{F}^f$, a generalization of Coulomb's friction law is now defined by the following principle.

Maximum dissipation principle. (p_t^f, p_o^f) is an optimal solution of the following maximization problem in the variables (q_t, q_o) :

$$(37) \quad \text{maximize} \quad \{-(q_t \dot{w}_t^f + q_o \dot{w}_o^f) : (q_t, q_o) \in \mathcal{F}^f(\mu^f p_n^f)\},$$

where μ^f is the friction coefficient and a superposed dot denotes the time derivative. The interpretation of this principle is that for prescribed tangential displacement rates $\dot{w}_t^f$ and $\dot{w}_o^f$ and the normal contact traction p_n^f , the tangential contact traction will adjust itself so as to give a maximum dissipation of energy. Coulomb's friction law is often stated in several equivalent forms, depending on the explicit representation of the friction function $\mathcal{F}^f$; see section 2.

The continuous version of our problem is now defined by (32)–(37), together with appropriate initial conditions. Our main concern in this work is a discrete version of this problem. We would like to show how a formal finite element discretization will produce such a formulation. To that end, we give a variational formulation of the problem.

Continuous Problem ($\mathcal{V}$). Find $(u^f \equiv (u_1^f, u_2^f, u_3^f), p_n^f, p_t^f, p_o^f) : [0, T] \rightarrow V \times K_n \times K_t$ satisfying initial conditions, $(p_t^f, p_o^f) \in \mathcal{F}^f(\mu^f p_n^f)$, and

$$(38) \quad a(u^f, v^f) + \int_{S_c} (p_n^f, p_t^f, p_o^f)' C^f v^f ds = f(v^f) \quad \forall v^f \in V,$$

$$(39) \quad \int_{S_c} (g_n^f - C_n^f u^f)(q_n^f - p_n^f) ds \geq 0 \quad \forall q_n^f \in K_n,$$

$$(40) \quad \int_{S_c} [(C_t^f \dot{u}^f)(q_t^f - p_t^f) + (C_o^f \dot{u}^f)(q_o^f - p_o^f)] ds \leq 0 \quad \forall (q_t^f, p_o^f) \in \mathcal{F}^f(\mu^f p_n^f).$$

Here, $[0, T]$ is the time interval under consideration, V is the space of vector fields on Ω which are zero on S_u , K_n is the set of nonnegative scalar fields on S_c , and K_t is the set of $\mathbb{R}^2$ -valued fields on S_c . The bilinear form $a(\cdot, \cdot)$ is well defined for $V \subset H^1(\Omega)$, and analysis shows that smooth solutions can generally not be expected. This implies that in a mathematically rigorous version of the above variational formulation, we need to interpret integrals over S_c as dual pairings between a trace space of $H^1(\Omega)$ and its dual. For the mathematical analysis of this and similar continuous contact problems, we refer to Kikuchi and Oden [29] or Hlaváček et al. [21]. However, since our goal is to present a formal finite element discretization, these questions are not central in the paper. The notations $(\cdot)_n$, $(\cdot)_t$, and $(\cdot)_o$ denote the first through third components of the vector $(\cdot) \in \mathbb{R}^3$. Furthermore, we have the bilinear form

$$a(u^f, v^f) = \int_{\Omega} E_{ijkl}^f \frac{\partial u_i^f}{\partial x_j} \frac{\partial u_k^f}{\partial x_l} d\Omega, \quad u^f, v^f \in V$$

and the force functional

$$f(v^f) = \int_{S_t} \bar{t}_i^f v_i^f ds + \int_{\Omega} \bar{f}_i^f v_i^f d\Omega, \quad v^f \in V.$$

It may be seen that, for sufficiently smooth solutions, problem ($\mathcal{V}$) is equivalent to (32)–(37); see, for instance, [28]. Equation (38) is simply the principle of virtual work. Inequality (39) is a global variational version of Signorini's contact conditions, while (40) is a global version of the maximum dissipation principle defining Coulomb's law of friction.

5.2. Finite element and time discretization. The finite element discretization of the forms $a(u^f, v^f)$ and $f(v^f)$ is classical and will not be considered in detail here. In short, these forms can be written as a bilinear and a linear function. Let u^{hf} be a finite dimensional approximation of u^f , represented by a vector $u \in \mathbb{R}^{n_u}$ of nodal values of u^{hf} , and similarly for the test function v^f . We then have

$$a(u^{hf}, v^{hf}) = u' K v, \quad f(v^{hf}) = f'_{\text{ext}} v,$$

where K is the symmetric n_u -by- n_u finite element *stiffness matrix* and $f_{\text{ext}} \in \mathbb{R}^{n_u}$ is a finite dimensional representation of the external forces f_i^f and t_i^f . Regarding the other terms of problem (V), i.e., those involving integrals over S_c , they will be treated by numerical integration. That is, the integral of a function f defined over S_c is approximated as follows:

$$(41) \quad \int_{S_c} f \, ds \approx \sum_{i=1}^{n_c} I^i f(x^i),$$

where I^i are integration weights, $x^i \equiv (x_1^i, x_2^i, x_3^i)$ are the integration points, and n_c is their number. Useful particular examples of (41) are those provided by the trapezoidal quadrature rule, Simpson's rule, and generalizations of these rules for the three-dimensional case. Note that although it seems that the most stable discretization schemes are such that integration points coincide with finite element nodal points, this is not required by the formal considerations given here. With the integration rule (41), (38) then results in

$$(42) \quad u' K v + \sum_{i=1}^{n_c} I^i \left(p_n^f(x^i), p_t^f(x^i), p_o^f(x^i) \right)' C^f(x^i) v^{hf}(x^i) = f'_{\text{ext}} v$$

for all $v^{hf} \in V^h$. Next, introduce force vectors

$$p_\alpha \equiv \{I^i p_n^f(x^i)\} \in \mathbb{R}^{n_c} \quad \text{for } \alpha = n, t, o$$

and define n_c -by- n_u matrices C_n , C_t , and C_o by means of the expressions

$$C_\alpha v \equiv \{(C^f(x^i) v^{hf}(x^i))_n\} \quad \text{for } \alpha = n, t, o.$$

With these notations, (42) becomes

$$(43) \quad u' K v + p_n' C_n v + p_t' C_t v + p_o' C_o v = f'_{\text{ext}} v \quad \forall v \in \mathbb{R}^{n_u},$$

which is equivalent to the equilibrium equation

$$(44) \quad K u + C_n' p_n + C_t' p_t + C_o' p_o = f_{\text{ext}}.$$

We now give a discrete approximation of (39). To give such an inequality one has to introduce an approximation of the constraint set K_n . To that end, we will require that $p_n^f \geq 0$ is satisfied at the integration points x^i , $i = 1, \dots, n_c$. The set K_n is then replaced by $\mathbb{R}_+^{n_c}$, i.e., the nonnegative orthant of $\mathbb{R}^{n_c}$. With the notations previously introduced, (39) now becomes a variational inequality which is equivalent to the following complementarity conditions:

$$(45) \quad p_n \geq 0, \quad w_n \geq 0, \quad p_n' w_n = 0,$$

where $w_n \equiv g_n - C_n u$, $g_n \equiv \{g_n^f(x^i)\}$.

The constraint set of inequality (40) is approximated similarly to K_n ; i.e., we require that this condition is satisfied at the integration points. In this context it is convenient to assume that the graph of $\mathcal{F}$ is a cone. We obtain the following discrete version of the maximum dissipation principle.

Discrete maximum dissipation principle. For $i = 1, \dots, n_c$, (p_{it}, p_{io}) is an optimal solution of the following maximization problem in the variables (q_{it}, q_{io}) :

$$(46) \quad \text{maximize} \quad \{-(q_{it}\dot{w}_{it} + q_{io}\dot{w}_{io}) : (q_{it}, q_{io}) \in \mathcal{F}_i(\mu_i p_{in})\},$$

where $\dot{w}_t \equiv -C_t \dot{u}$, $\dot{w}_o \equiv -C_o \dot{u}$, $(q_{it}, q_{io}) \equiv I^i(q_t^f(x^i), q_o^f(x^i))$, and $\mathcal{F}_i(\mu_i p_{in}) \equiv \mathcal{F}^f(I^i \mu^f(x^i) p_n^f(x^i))$.

In summary, the discrete frictional contact problem is the following.

Discrete Problem ($\mathcal{D}$). Find $(u, p_n, p_t, p_o) : [0, T] \rightarrow \mathbb{R}^{n_u} \times \mathbb{R}_+^{n_c} \times \mathbb{R}^{2n_c}$ satisfying initial conditions and (44), (45), and (46).

The time incremental problem. Let the time interval $[0, T]$ be divided into subintervals. One such subinterval is $(t_\ell, t_{\ell+1})$. At time t_ℓ the vectors u , p_n , p_t , and p_o are assumed to be known and are denoted $\bar{u}$, $\bar{p}_n$, $\bar{p}_t$, and $\bar{p}_o$. The force f_{ext} at time $t_{\ell+1}$ is prescribed and is denoted $f_{\text{ext}}(t_{\ell+1})$. Furthermore, the time derivative in (46) is approximated by a backward finite difference

$$\dot{u} \approx \frac{u - \bar{u}}{t_{\ell+1} - t_\ell}.$$

The discrete maximum dissipation principle (46) then becomes a discrete maximum work principle.

Discrete maximum work principle. For $i = 1, \dots, n_c$, (p_{it}, p_{io}) is an optimal solution of the following maximization problem in the variables (q_{it}, q_{io}) :

$$(47) \quad \text{maximize} \quad \{q_{it}(C_t(u - \bar{u}))_i + q_{io}(C_o(u - \bar{u}))_i : (q_{it}, q_{io}) \in \mathcal{F}_i(\mu_i p_{in})\}.$$

The time incremental discrete problem at time $t_{\ell+1}$ is the following.

Incremental problem ($\mathcal{D}_{\text{incr}}$). Find $(u, p_n, p_t, p_o) \in \mathbb{R}^{n_u} \times \mathbb{R}_+^{n_c} \times \mathbb{R}^{2n_c}$ satisfying (44) with $f_{\text{ext}} \equiv f_{\text{ext}}(t_{\ell+1})$, (45), and (47).

The last problem is the one treated in this paper.

REFERENCES

- [1] P. ALART, *Critères d'injectivité et de surjectivité pour certaines applications de $\mathbb{R}^n$ dans lui-même; application à la mécanique du contact*, Math. Modelling Numer. Anal., 27 (1993), pp. 203–222.
- [2] A. ALART, F. LEBON, F. QUITTAU, AND K. REY, *Frictional contact problem in elastostatics: Revisiting the uniqueness condition*, in Contact Mechanics, M. Raous, M. Jean, and J. J. Moreau, eds., Plenum Press, New York, 1995, pp. 63–70.
- [3] A. M. AL-FAHED AND P. D. PANAGIOTOPOULOS, *Multifingered frictional robot grippers: A new type of numerical implementation*, Comput. & Structures, 42 (1992), pp. 555–562.
- [4] A. M. AL-FAHED, G. E. STAVROLAKIS, AND P. D. PANAGIOTOPOULOS, *Hard and soft fingered robot grippers. The linear complementarity approach*, Z. Angew. Math. Mech., 71 (1991), pp. 257–265.
- [5] L.-E. ANDERSSON, *A quasistatic frictional problem with normal compliance*, Nonlinear Anal., 16 (1991), pp. 347–369.
- [6] L.-E. ANDERSSON, *A global existence result for a quasistatic contact problem with friction*, Adv. Math. Sci. Appl., 5 (1995), pp. 249–286.
- [7] G. BONFANTI, *A noncoercive friction problem with tangential applied forces in three dimensions*, Boll. Un. Mat. Ital. B (7), 7 (1993), pp. 149–165.
- [8] D. CHAN AND J. S. PANG, *The generalized quasi-variational inequality problem*, Math. Oper. Res., 7 (1982), pp. 211–222.

- [9] P. W. CHRISTENSEN, A. KLARBRING, J. S. PANG, AND N. STROMBERG, *Formulation and comparison of algorithms for frictional contact problems*, Internat. J. Num. Methods Engrg., to appear.
- [10] P. W. CHRISTENSEN AND J. S. PANG, *Frictional contact algorithms based on semismooth Newton methods*, in Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Method, M. Fukushima and L. Qi, eds., Kluwer Academic Publishers, Norwell, MA, to appear.
- [11] M. COCU, *Existence of solutions of Signorini problems with friction*, Internat. J. Engrg. Sci., 22 (1984), pp. 567–575.
- [12] R. W. COTTLE, J. S. PANG, AND R. E. STONE, *The Linear Complementarity Problem*, Academic Press, Boston, MA, 1992.
- [13] L. DEMKOWICZ AND J. T. ODEN, *On some existence and uniqueness results in contact problems with nonlocal friction*, Nonlinear Anal., 6 (1982), pp. 1075–1093.
- [14] I. DOUDOUMIS, E. MITSOPOULOU, AND N. CHARALAMBAKIS, *The influence of the friction coefficients on the uniqueness of the solution of the unilateral contact problem*, in Contact Mechanics, M. Raous, M. Jean, and J. J. Moreau, eds., Plenum Press, New York, 1995, pp. 63–70.
- [15] G. DUVAUT, *Equilibre d'un solide élastique avec contact unilatéral et frottement de Coulomb*, Comptes rendues Academie Sciences, Paris/A, 290 (1980), pp. 263–265.
- [16] G. DUVAUT AND J. L. LIONS, *Inequalities in Mechanics and Physics*, Springer, Berlin, 1976.
- [17] G. FICHERA, *Boundary value problems in elasticity with unilateral constraints*, in Encyclopedia of Physics, Vol. VI a/2, S. Flügge, ed., Springer-Verlag, Berlin, 1972.
- [18] F. GASTALDI, *Remarks on a Noncoercive Contact Problem with Friction in Elastostatics*, Tech. report 649, Istituto di analisi numerica, University of Pavia, Pavia, 1988.
- [19] F. GASTALDI, M. D. P. MONTEIRO MARQUES, AND J. A. C. MARTINS, *A two degree-of-freedom quasistatic contact problem with viscous damping*, Adv. Math. Sci. Appl., submitted.
- [20] F. GASTALDI, M. D. P. MONTEIRO MARQUES, AND J. A. C. MARTINS, *Mathematical analysis of a two degree-of-freedom frictional contact problem with discontinuous solutions*, Nonlinear Differential Equations Appl., submitted.
- [21] I. HLAVÁČEK, J. HASLINGER, J. NEČAS, AND J. LOVÍČEK, *Numerical Solution of Variational Inequalities*, Springer Series in Applied Mathematical Sciences 66, Springer-Verlag, New York, 1988.
- [22] V. JANOVSKÝ, *Catastrophic Features of Coulomb Friction Model*, Tech. report KNM-0105044/80, Department of Numerical Analysis, Charles University, Prague, Czech Republic, 1980.
- [23] V. JANOVSKÝ, *Catastrophic features of Coulomb friction model*, in The Mathematics of Finite Elements and Applications, J. R. Whiteman, ed., Academic Press, London, 1981, pp. 259–264.
- [24] J. JARUŠEK, *Contact problems with friction. Coercive case*, Czechoslovak Math. J., 33 (1983), pp. 237–261.
- [25] J. JARUŠEK, *Contact problems with friction. Semicoercive case*, Czechoslovak Math. J., 34 (1984), pp. 619–629.
- [26] J. H. JELLET, *A Treatise on the Theory of Friction*, Macmillan and Co., London, 1872.
- [27] J. J. KALKER, *Three-dimensional elastic bodies in rolling contact*, Kluwer, Dordrecht, 1990.
- [28] Y. KATO, *Signorini's problem with friction in linear elasticity*, Japan J. Appl. Math., 4 (1987), pp. 237–268.
- [29] N. KIKUCHI AND J. T. ODEN, *Contact Problems in Elasticity: A Study of Variational Inequalities and Finite Element Methods*, SIAM Studies Appl. Math., Philadelphia, PA, 1988.
- [30] A. KLARBRING, *Contact Problems with Friction—Using a Finite Dimensional Description and the Theory of Linear Complementarity*, Thesis no. 20, LIU-TEK-LIC-1984:3, Linköping Studies Sci. Tech., Linköping Institute of Technology, Linköping, Sweden, 1984.
- [31] A. KLARBRING, *A mathematical programming approach to three-dimensional contact problems with friction*, Comput. Methods Appl. Mech. Engrg, 58 (1986), pp. 175–200.
- [32] A. KLARBRING, *Contact problems with friction by linear complementarity*, in Unilateral Problems in Structural Analysis, Vol. 2 (CISM Courses and Lectures, No. 304), G. Del Piero and F. Maceri, eds., Springer, Wien, 1987, pp. 197–219.
- [33] A. KLARBRING, *Derivation and analysis of rate boundary-value problems with friction*, European J. Mech. A Solids, 1 (1990), pp. 211–226.
- [34] A. KLARBRING, *Examples of non-uniqueness and non-existence of solutions to quasistatic contact problems with friction*, Ingenieur-Archiv, 60 (1990), pp. 529–541.
- [35] A. KLARBRING, *Mathematical programming and augmented Lagrangian methods for frictional contact problems*, in Proc. Contact Mechanics International Symposium, A. Curnier, ed., Presses Polytechniques et Universitaires Romandes, Lausanne, Switzerland, 1992, pp. 409–422.

- [36] A. KLARBRING, *Mathematical programming in contact problems*, in Computational Methods in Contact Mechanics, M. H. Aliabadi and C. A. Brebbia, eds., Computational Mechanics Publications, Southampton, UK, 1993, pp. 233–263.
- [37] A. KLARBRING AND G. BJÖRKMAN, *A mathematical programming approach to contact problems with friction and varying contact surface*, Comput. & Structures, 30 (1988), pp. 1185–1198.
- [38] A. KLARBRING, A. MIKELIĆ, AND M. SHILLOR, *Frictional contact problems with normal compliance*, Internat. J. Engrg. Sci., 26 (1988), pp. 811–832.
- [39] A. KLARBRING, A. MIKELIĆ, AND M. SHILLOR, *On friction problems with normal compliance*, Nonlinear Anal., 13 (1989), pp. 935–955.
- [40] A. KLARBRING, A. MIKELIĆ, AND M. SHILLOR, *A global existence result for the quasistatic frictional contact problem with normal compliance*, Internat. Ser. Numer. Math. 101, Birkhäuser, Basel, 1991, pp. 85–111.
- [41] J. KYPARISIS, *On uniqueness of Kuhn-Tucker multipliers in nonlinear programming*, Math. Programming, 32 (1985), pp. 242–246.
- [42] O. L. MANGASARIAN, *A simple characterization of solution sets of convex programs*, Oper. Res. Lett., 7 (1988), pp. 21–26.
- [43] J. A. C. MARTINS, M. D. P. MONTEIRO MARQUES, AND F. GASTALDI, *On an example of non-existence of solutions to a quasistatic frictional contact problem*, European J. Mech. A Solids, 13 (1994), pp. 113–133.
- [44] J. A. C. MARTINS AND J. T. ODEN, *Existence and uniqueness results for dynamic contact problems with nonlinear normal and friction interface laws*, Nonlinear Anal., 11 (1987), pp. 407–428.
- [45] J. A. C. MARTINS, F. M. F. SIMÕES, F. GASTALDI, AND M. D. P. MONTEIRO MARQUES, *Dissipative graph solutions for a 2 degree-of-freedom quasistatic frictional contact problem*, Internat. J. Engrg. Sci., 33 (1995), pp. 1959–1986.
- [46] M. D. P. MONTEIRO MARQUES, *Differential Inclusions in Nonsmooth Mechanical Problems: Shocks and Dry Friction*, Birkhäuser, Basel, 1993.
- [47] J. J. MOREAU, *Unilateral contact and dry friction in finite freedom dynamics*, in Nonsmooth Mechanics and Applications, J. J. Moreau and P. D. Panagiotopoulos, eds., Springer-Verlag, Berlin, 1988, pp. 1–81.
- [48] J. NEČAS, J. JARUŠEK, AND J. HASLINGER, *On the solution of the variational inequality to the Signorini problem with small friction*, Boll. Un. Mat. Ital. B(5), 17 (1980), no. 2, pp. 796–811.
- [49] P. D. PANAGIOTOPOULOS AND A. M. AL-FAHED, *Robot hand grasping and related problems: Optimal control and identification*, Internat. J. Robotics Res., 13 (1994), pp. 127–136.
- [50] J. S. PANG AND J. C. TRINKLE, *Complementarity formulations and existence of solutions of multi-rigid-body contact problems with Coulomb friction*, Math. Programming, 73 (1996), pp. 199–226.
- [51] J. S. PANG, J. C. TRINKLE, AND G. LO, *A complementarity approach to a quasi-static multi-rigid-body contact problem*, Comput. Optim. Appl., 5 (1996), pp. 139–154.
- [52] M. RAOUS, M. JEAN, AND J. J. MOREAU, EDS., *Contact Mechanics*, Plenum Press, New York, 1995.
- [53] S. M. ROBINSON, *Some continuity properties of polyhedral multifunctions*, Math. Programming Study, 14 (1981), pp. 206–214.
- [54] J. C. TRINKLE, J. S. PANG, S. SUDARSKY, AND G. LO, *On dynamic multi-rigid-body contact problems with Coulomb friction*, Z. Angew. Math. Mech., 77 (1997), pp. 267–279.
- [55] L. K. VY AND D. D. ANG, *Frictional contact of an elastic body with a rigid support*, Nonlinear Anal., 25 (1995), pp. 339–343.

AN UNCONSTRAINED CONVEX PROGRAMMING APPROACH TO LINEAR SEMI-INFINITE PROGRAMMING*

CHIH-JEN LIN[†], SHU-CHERNG FANG[‡], AND SOON-YI WU[§]

Abstract. In this paper, an unconstrained convex programming dual approach for solving a class of linear semi-infinite programming problems is proposed. Both primal and dual convergence results are established under some basic assumptions. Numerical examples are also included to illustrate this approach.

Key words. semi-infinite programming, linear programming, convex programming, entropy optimization

AMS subject classifications. 90C05, 90C34, 49M35

PII. S105262349427621X

1. Introduction. Many linear semi-infinite programming problems including the L_1 and Chebyshev approximation problems [14, 15] appear in the following “dual form.”

Program (D).

$$(1.1) \quad \begin{aligned} & \text{Max } b^T w \\ & \text{s.t. } a(t)^T w \leq c(t) \quad \forall t \in T, \end{aligned}$$

where $b, w \in R^m$, T is a compact set in R^n , $a(t)^T = (a_1(t), \dots, a_m(t))$, and $c(t)$ and $a_j(t)$, $j = 1, \dots, m$, are continuous functions defined on T .

A corresponding “primal form” linear semi-infinite programming problem can be represented as follows.

Program (P).

$$(1.2) \quad \begin{aligned} & \text{Min } \int_T c(t)x(t)d\lambda(t) \\ & \text{s.t. } \int_T a_j(t)x(t)d\lambda(t) = b_j, \quad j = 1, \dots, m, \\ & x(t) \in L_1^+(T, \lambda), \end{aligned}$$

where $\lambda(t)$ is the Lebesgue measure, a particular regular Borel measure, on T and $L_1^+(T, \lambda) = \{x(t) \mid x(t) \text{ is measurable on } T, x(t) \geq 0 \text{ almost everywhere (a.e.) with respect to } \lambda, \text{ and } \int_T |x(t)|d\lambda(t) < +\infty\}$. To simplify our expressions, equation (1.2) will be denoted by $\int_T a(t)x(t)d\lambda(t) = b$.

The duality theory relating Programs (P) and (D) can be found in [17]. Under certain conditions, a strong duality theorem holds. According to [17], there exist three

*Received by the editors October 24, 1994; accepted for publication (in revised form) March 4, 1997.

<http://www.siam.org/journals/siopt/8-2/27621.html>

[†]Department of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, MI 48109 (cjlin@engin.umich.edu).

[‡]Operations Research and Industrial Engineering, North Carolina State University, Raleigh, NC 27695-7913 (fang@eos.ncsu.edu). The research of this author was supported in part by the 1994 NCSC-Cray Research Grant.

[§]Institute of Applied Mathematics, National Cheng-Kung University, Tainan, Taiwan, R.O.C. (soonyi@mail.ncku.edu.tw).

“basic” solution approaches, namely, the exchange methods, discretization methods, and methods based on local reduction. All these methods usually replace Program (D) by a sequence of finite linear programming problems for approximation and require a search procedure for global optimization on T . This could be very costly, especially when T has higher dimensionality [13, 15, 16, 18].

In this paper, we propose a solution approach which identifies an optimal solution of Program (D) as a limit point of the solutions of a sequence of finite dimensional unconstrained convex programming problems. No global optimization is required in the proposed approach, but instead, a multidimensional integration is required. The basic ideas are introduced in section 2, and main results are given in section 3. Primal convergence results are derived in section 4, and numerical examples are given in section 5. The last section includes some concluding remarks.

2. Basic ideas. Following [11], given $\mu > 0$, we perturb Program (P) by adding an entropic term $\mu \int_T x(t) \log x(t) d\lambda(t)$ to its objective to form a perturbed problem as follows.

Program (P_μ).

$$(2.1) \quad \begin{aligned} & \text{Min} \int_T c(t)x(t)d\lambda(t) + \mu \int_T x(t) \log x(t)d\lambda(t) \\ & \text{s.t.} \quad \int_T a(t)x(t)d\lambda(t) = b, \\ & \quad x(t) \in L_1^+(T, \lambda). \end{aligned}$$

The conjugate dual [4, 20] of Program (P_μ) is given as follows.

Program (D_μ).

$$(2.2) \quad \begin{aligned} & \text{Max} \quad b^T w - \mu \int_T e^{\frac{a(t)^T w - c(t)}{\mu} - 1} d\lambda(t) \\ & \text{s.t.} \quad w \in R^m. \end{aligned}$$

When its optimal solution exists, we denote $x_\mu^*(t)$ as an optimal solution of Program (P_μ), with an optimal objective value $v(P_\mu)$. Similarly, w_μ^* denotes an optimal solution of Program (D_μ) with an optimal objective value $v(D_\mu)$.

Note that (D_μ) is an unconstrained convex program which can be treated by various numerical techniques [8]. This provides an alternative approach even though the bottleneck becomes the numerical integration of a multidimensional integration over a compact set T in R^n . Compared to Program (D), solving (D_μ) is like a penalty function method [5] with an exponential penalty term

$$\mu \int_T e^{\frac{a(t)^T w - c(t)}{\mu} - 1} d\lambda(t).$$

Other penalty function methods used for semi-infinite programming problems can be found in [6, 19]. In this paper we focus on using the exponential penalty function for linear semi-infinite programming as an extension of the methods developed for solving linear programming problems [10, 12].

In order to show that the proposed approach works, we need to address three basic issues [1].

- (i) The existence of an optimal solution w_μ^* to Program (D_μ) $\forall \mu > 0$.
- (ii) The existence of a compact set and a positive constant Θ such that w_μ^* lies in the compact set $\forall 0 < \mu < \Theta$.

- (iii) If a sequence $\{w_{\mu_i}^*\}$ converges to w^* as $\mu_i \rightarrow 0$, then w^* is an optimal solution of (D) .

In general, issues (i) and (ii) are much more difficult to deal with than the third one. In this paper, we refer to Borwein and Lewis [2] for issue (i) and resolve the other two in section 3. Related results for the finite dimensional case can be found in [9].

Throughout this paper, two commonly seen assumptions are made.

Assumption (A1). Program (D) has an interior feasible solution, i.e., $\{w \in R^m \mid a(t)^T w - c(t) < 0 \ \forall t \in T\} \neq \emptyset$.

Assumption (A2). The primal constraint qualification (PCQ) holds; i.e., there exists $\hat{x}$ in $L_1(T, \lambda) \triangleq \{x \mid x(t) \text{ is measurable on } T \text{ and } \int_T |x(t)| d\lambda(t) < +\infty\}$ such that $\hat{x}(t) > 0$ a.e. with respect to the Lebesgue measure, $\int_T c(t) \hat{x}(t) d\lambda(t) + \mu \int_T \hat{x}(t) \log \hat{x}(t) d\lambda(t) < \infty$, and $\int_T a(t) \hat{x}(t) d\lambda(t) = b$.

With Assumption (A2), Theorem 4.2 of [2] resolves issue (i) and assures that both Programs (P_μ) and (D_μ) attain optimality with $v(P_\mu) = v(D_\mu)$. Moreover, if w_μ^* is optimal to Program (D_μ) , then

$$(2.3) \quad x_\mu^*(t) = e^{\frac{a(t)^T w_\mu^* - c(t)}{\mu} - 1} \quad \forall t \in T$$

is the unique optimal solution of Program (P_μ) . This provides a dual-to-primal conversion formula. Also note that since the objective function of Program (D_μ) is strictly concave, under Assumption (A2), its optimal solution w_μ^* can be obtained by solving the following first order condition:

$$(2.4) \quad \int_T a(t) e^{\frac{a(t)^T w - c(t)}{\mu} - 1} d\lambda(t) = b.$$

3. Main results. The objective of this section is to address issues (ii) and (iii). Since issue (ii) is more difficult to handle, we start with issue (iii).

THEOREM 3.1. *If $\{w_{\mu_i}^*\} \rightarrow w^*$ as $\mu_i \rightarrow 0$, then w^* is an optimal solution of Program (D) .*

Proof. Feasibility. Under Assumption (A1), Program (D) has a feasible solution $\bar{w}$ with

$$(3.1) \quad b^T \bar{w} - \mu_i \int_T e^{\frac{a(t)^T \bar{w} - c(t)}{\mu_i} - 1} d\lambda(t) \leq b^T w_{\mu_i}^* - \mu_i \int_T e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t).$$

If w^* is not feasible for Program (D) , then there exists $\bar{t} \in T$ such that $a(\bar{t})^T w^* - c(\bar{t}) > 0$. Since T is a compact set in R^n and $a_j(t), c(t)$ are continuous on T , there exists a neighborhood $N_\delta(\bar{t})$ of $\bar{t}$ and an $\epsilon > 0$ such that $a(t)^T w^* - c(t) \geq \epsilon$ for $t \in N_\delta(\bar{t})$.

By using L'Hôpital's rule, we have

$$(3.2) \quad \mu_i \int_{N_\delta(\bar{t})} e^{\frac{\epsilon}{\mu_i}} d\lambda(t) \rightarrow \infty \text{ as } \mu_i \rightarrow 0.$$

Hence the right-hand side of equation (3.1) approaches $-\infty$ as $\mu_i \rightarrow 0$. Moreover, since $a(t)^T \bar{w} - c(t) \leq 0 \ \forall t \in T$, we have

$$(3.3) \quad 0 \leq \lim_{\mu_i \rightarrow 0} \mu_i \int_T e^{\frac{a(t)^T \bar{w} - c(t)}{\mu_i} - 1} d\lambda(t) \leq \lim_{\mu_i \rightarrow 0} \mu_i \int_T e^{-1} d\lambda(t) = 0.$$

Consequently, the left-hand side of equation (3.1) approaches $b^T \bar{w}$ as $\mu_i \rightarrow 0$. This causes a contradiction. Therefore, w^* must be feasible for Program (D) .

Optimality. If w^* is not an optimal solution of Program (D), then we can find a feasible solution $\bar{w}$ with $b^T \bar{w} > b^T w^*$. Since $w_{\mu_i}^*$ is optimal for Program (D_{μ_i}) , we know

$$b^T \bar{w} - \mu_i \int_T e^{\frac{a(t)^T \bar{w} - c(t)}{\mu_i} - 1} d\lambda(t) \leq b^T w_{\mu_i}^* - \mu_i \int_T e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) \leq b^T w_{\mu_i}^*.$$

With the same reason we had for equation (3.3) as $\mu_i \rightarrow 0$, we have $b^T \bar{w} \leq b^T w^*$. This again causes a contradiction and hence completes the proof. $\square$

As a direct consequence, we have the next result.

COROLLARY 3.2. *If $\{w_{\mu_i}^*\} \rightarrow w^*$ as $\mu_i \rightarrow 0$, then*

$$(3.4) \quad \lim_{\mu_i \rightarrow 0} \mu_i \int_T e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) = 0.$$

Proof. Since

$$(3.5) \quad \begin{aligned} & b^T w^* - \mu_i \int_T e^{\frac{a(t)^T w^* - c(t)}{\mu_i} - 1} d\lambda(t) \\ & \leq b^T w_{\mu_i}^* - \mu_i \int_T e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) \\ & \leq b^T w_{\mu_i}^* \end{aligned}$$

as $\mu_i \rightarrow 0$, we have

$$b^T w^* \leq b^T w^* - \lim_{\mu_i \rightarrow 0} \mu_i \int_T e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) \leq b^T w^*.$$

Therefore,

$$\lim_{\mu_i \rightarrow 0} \mu_i \int_T e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) = 0. \quad \square$$

To handle issue (ii), we make an additional assumption.

Assumption (A3) (bounded level set assumption). There exists a constant L such that $\{w \mid a(t)^T w - c(t) \leq 0\} \cap \{w \mid b^T w \geq L\}$ is nonempty and bounded.

Note that Assumption (A3) implies that Program (D) is solvable. We shall keep using this constant L throughout the rest of the paper.

Given that $\epsilon > 0$, $l > 0$, λ is the Lebesgue measure on T , and $\bar{w} \in R^m$, we define the following notations:

1. $\lambda(\bar{w}, \epsilon) = \lambda(\{t \mid a(t)^T \bar{w} - c(t) > \epsilon\})$,
2. $B = \{w \in R^m \mid b^T w \geq L\}$,
3. $S(l) = \{w \in R^m \mid a(t)^T w - c(t) \leq l \ \forall t \in T\}$,
4. $\bar{S}(l) = \{w \in S(l) \mid \text{there exists } \bar{t} \in T \text{ such that } a(\bar{t})^T w - c(\bar{t}) = l\}$.

Consequently, the feasible region of the original problem $F \triangleq \{w \in R^m \mid a(t)^T w - c(t) \leq 0 \ \forall t \in T\}$ becomes $S(0)$ and Assumption (A3) becomes that $S(0) \cap B$ is nonempty and bounded.

Now we prove that, under some conditions, there exist $\Theta > 0$ and $\bar{l} > 0$ such that w_{μ}^* lies in a compact set $S(\bar{l}) \cap B \ \forall 0 < \mu \leq \Theta$. In this way, we have a convergent subsequence which goes to an optimal solution w^* .

The basic idea is to use $\lambda(w_\mu^*, \epsilon)$ as a measure of those t 's at which the constraint $a(t)^T w - c(t) \leq 0$ is violated. Then, if w_μ^* lies outside $S(\bar{l}) \cap B$, we can prove that $\lambda(w_\mu^*, \epsilon)$ is bounded above by zero, which causes a violation of the first order necessary condition

$$\int_T a(t) e^{\frac{a(t)^T w_\mu^* - c(t)}{\mu_i} - 1} d\lambda(t) = b$$

as $\mu_i \rightarrow 0$.

THEOREM 3.3. *Under Assumption (A3), there exists $\bar{l} > 0$ such that $S(\bar{l}) \cap B$ is compact.*

Proof. It is easy to see that $S(l) \cap B$ is closed for $l > 0$. Hence, we only have to prove that there exists $\bar{l} > 0$ such that $S(\bar{l}) \cap B$ is bounded.

If our claim is not true, there must exist a sequence $\{l_i\}$, with $\lim_{i \rightarrow \infty} l_i = 0$, such that $S(l_i) \cap B$ is unbounded. Hence, we can select an unbounded sequence $\{w_{l_i}\}$ with $w_{l_i} \in S(l_i) \cap B$ for each l_i . It is obvious that $\{w_{l_i} / \|w_{l_i}\|\}$ is a bounded sequence, thus we can find a subsequence $\{k_i\}$ of $\{l_i\}$ such that $\{w_{k_i} / \|w_{k_i}\|\}$ converges to a point, say $\hat{w}$, as $k_i \rightarrow 0$.

Since $S(0) \cap B$ is nonempty, we can find at least one $w_0 \in S(0) \cap B$. For any $\beta > 0$, since $S(k_i) \cap B$ is convex and $w_0, w_{k_i} \in S(k_i) \cap B$, we have

$$\left(1 - \frac{\beta}{\|w_{k_i}\|}\right) w_0 + \left(\frac{\beta}{\|w_{k_i}\|}\right) w_{k_i} \in S(k_i) \cap B$$

for k_i being sufficiently small.

Equivalently, for any $t \in T$,

$$a(t)^T \left(\left(1 - \frac{\beta}{\|w_{k_i}\|}\right) w_0 + \frac{\beta}{\|w_{k_i}\|} w_{k_i} \right) - c(t) \leq k_i.$$

As $k_i \rightarrow 0$, $a(t)^T (w_0 + \beta \hat{w}) - c(t) \leq 0$ and $b^T (w_0 + \beta \hat{w}) \geq L \forall t \in T$. Therefore, for any $\beta > 0$, $w_0 + \beta \hat{w} \in S(0) \cap B$. Note that $\hat{w} \neq 0$. This contradicts Assumption (A3), under which $S(0) \cap B$ is bounded. $\square$

By using the constant $\bar{l} > 0$ obtained in Theorem 3.3, we can derive the following results.

LEMMA 3.4. *Given $\epsilon > 0$ and $\bar{l} > \epsilon > 0$, there exists $\delta > 0$ such that $\lambda(\bar{w}, \epsilon) \geq \delta \forall \bar{w} \in \bar{S}(\bar{l}) \cap B$.*

Proof. Since $\bar{S}(\bar{l}) \cap B$ is a closed subset of $S(\bar{l}) \cap B$, it is compact. With $\lambda(\bar{w}, \epsilon) \geq 0$, we have

$$(3.6) \quad \delta \triangleq \inf\{\lambda(\bar{w}, \epsilon) \mid \bar{w} \in \bar{S}(\bar{l}) \cap B\} \geq 0.$$

If $\delta = 0$, then there exists a sequence $\{\bar{w}_i \in \bar{S}(\bar{l}) \cap B\}$ such that $\lim_{i \rightarrow \infty} \lambda(\bar{w}_i, \epsilon) = 0$. Since $\bar{S}(\bar{l}) \cap B$ is compact, there exists a subsequence which converges to a point $\hat{w}$ with $\lambda(\hat{w}, \epsilon) = 0$. Hence,

$$(3.7) \quad a(t)^T \hat{w} - c(t) \leq \epsilon \text{ a.e. in } T.$$

Remembering that $a_j(t)$ and $c(t)$ are continuous on T , we know $a(t)^T \hat{w} - c(t) \leq \epsilon \forall t \in T$.

However, since $\hat{w} \in \bar{S}(\bar{l}) \cap B$, from its definition, there exists $\bar{t} \in T$ such that $a(\bar{t})^T \hat{w} - c(\bar{t}) = \bar{l} > \epsilon$. This contradicts equation (3.7). Thus we have $\delta > 0$, and the proof follows. $\square$

LEMMA 3.5. *Given any $w_0 \in S(0) \cap B$ and $w_1 \in B \setminus S(\bar{l})$, the line segment $\overline{w_0 w_1}$ intersects $\bar{S}(\bar{l}) \cap B$ at exactly one point.*

Proof. Noting that $w_0 \in S(0) \cap B \subseteq F$, we have

$$(3.8) \quad a(t)^T w_0 - c(t) \leq 0 \quad \forall t \in T.$$

Since $w_1 \in \{w \in R^m \mid b^T w \geq L\} \setminus S(\bar{l}) \cap B$, there exists at least one $\bar{t} \in T$ such that

$$(3.9) \quad a(\bar{t})^T w_1 - c(\bar{t}) > \bar{l}.$$

Note that for $t \in T$ with $0 \leq \alpha \leq 1$,

$$(3.10) \quad \begin{aligned} & a(t)^T (w_0 + \alpha(w_1 - w_0)) \\ &= (1 - \alpha)(a(t)^T w_0 - c(t)) + \alpha(a(t)^T w_1 - c(t)) \\ &\leq \alpha(a(t)^T w_1 - c(t)). \end{aligned}$$

Define

$$(3.11) \quad \bar{\alpha} \triangleq \min\{\alpha > 0 \mid a(\bar{t})^T (w_0 + \alpha(w_1 - w_0)) - c(\bar{t}) \geq \bar{l} \text{ for some } \bar{t} \in T\}.$$

Together with (3.10), we know such $\bar{\alpha}$ exists and $\bar{\alpha} > 0$.

Note that $b^T(w_0 + \bar{\alpha}(w_1 - w_0)) \geq L$. We claim that $w_0 + \bar{\alpha}(w_1 - w_0) \in \bar{S}(\bar{l}) \cap B$. Otherwise there are two possible cases.

Case 1. There exists $\bar{t} \in T$ such that $a(\bar{t})^T (w_0 + \bar{\alpha}(w_1 - w_0)) - c(\bar{t}) > \bar{l}$. In this case, we can find a small number $\gamma > 0$ such that $a(\bar{t})^T (w_0 + (\bar{\alpha} - \gamma)(w_1 - w_0)) - c(\bar{t}) \geq \bar{l}$. This contradicts the definition of $\bar{\alpha}$.

Case 2. For all $t \in T$, $a(t)^T (w_0 + \bar{\alpha}(w_1 - w_0)) - c(t) < \bar{l}$. In this case, we can find a small number $\beta > 0$ such that

$$(3.12) \quad a(t)^T (w_0 + (\bar{\alpha} + \beta)(w_1 - w_0)) - c(t) < \bar{l} \quad \forall t \in T.$$

This again contradicts the definition of $\bar{\alpha}$.

Hence, $w_0 + \bar{\alpha}(w_1 - w_0) \in \bar{S}(\bar{l}) \cap B$. We now prove that $w_0 + \bar{\alpha}(w_1 - w_0)$ is the only point where $\overline{w_0 w_1}$ intersects $\bar{S}(\bar{l}) \cap B$.

Since $w_0 + \bar{\alpha}(w_1 - w_0) \in \bar{S}(\bar{l}) \cap B$, we know that there exists $\bar{t} \in T$ such that

$$(3.13) \quad a(\bar{t})^T (w_0 + \bar{\alpha}(w_1 - w_0)) - c(\bar{t}) = \bar{l}.$$

By equation (3.8), we have $a(\bar{t})^T w_0 - c(\bar{t}) \leq 0$. Combining with equation (3.13), we know $a(\bar{t})^T (\bar{\alpha}(w_1 - w_0)) > 0$. If there exists another intersection point, say $w_0 + \bar{\bar{\alpha}}(w_1 - w_0)$, then by the definition of $\bar{\alpha}$, we see $\bar{\bar{\alpha}} > \bar{\alpha}$ and $a(\bar{t})^T (w_0 + \bar{\bar{\alpha}}(w_1 - w_0)) - c(\bar{t}) > \bar{l}$. Hence, $w_0 + \bar{\bar{\alpha}}(w_1 - w_0)$ is not in $\bar{S}(\bar{l}) \cap B$ and this causes a contradiction. $\square$

LEMMA 3.6. *Given $\epsilon > 0$ and $\bar{l} > \epsilon > 0$, if $w_1 \in B \setminus S(\bar{l})$, then $\lambda(w_1, \epsilon) \geq \delta$, where $\delta > 0$ is given by Lemma 3.4.*

Proof. Take $w_0 \in S(0) \cap B$ and let $\hat{w}$ be the intersection point of $\overline{w_0 w_1}$ and $\bar{S}(\bar{l}) \cap B$. We can find $\alpha > 1$ such that

$$(3.14) \quad w_1 = w_0 + \alpha(\hat{w} - w_0).$$

Note that $1 - \alpha < 0$ and $a(t)^T w_0 - c(t) \leq 0 \quad \forall t \in T$. Therefore for each t with $a(t)^T \hat{w} - c(t) > \epsilon$, we have

$$(3.15) \quad \begin{aligned} & a(t)^T w_1 - c(t) \\ &= (1 - \alpha)(a(t)^T w_0 - c(t)) + \alpha(a(t)^T \hat{w} - c(t)) \\ &\geq \alpha(a(t)^T \hat{w} - c(t)) \\ &> \epsilon. \end{aligned}$$

By Lemma 3.4, we further have

$$\lambda(w_1, \epsilon) \geq \lambda(\hat{w}, \epsilon) \geq \delta. \quad \square$$

The next lemma shows that $b^T w_\mu^* > L$ if μ is sufficiently small.

LEMMA 3.7. *There exists $\Theta_1 > 0$ such that $b^T w_\mu^* \geq L$ for $0 < \mu < \Theta_1$.*

Proof. By Assumption (A3), we can find a point $\bar{w}$ that is feasible for Program (D) and $b^T \bar{w} > L$. Then,

$$\begin{aligned} b^T w_\mu^* &= v(D_\mu) + \mu \int_T e^{\frac{a(t)^T w_\mu^* - c(t)}{\mu} - 1} d\lambda(t) \\ (3.16) \quad &\geq b^T \bar{w} - \mu \int_T e^{\frac{a(t)^T \bar{w} - c(t)}{\mu} - 1} d\lambda(t) + \mu \int_T e^{\frac{a(t)^T w_\mu^* - c(t)}{\mu} - 1} d\lambda(t) \\ &\geq b^T \bar{w} - \mu \int_T e^{\frac{a(t)^T \bar{w} - c(t)}{\mu} - 1} d\lambda(t). \end{aligned}$$

Since $\bar{w}$ is feasible for (D), $\lim_{\mu \rightarrow 0} \mu \int_T e^{\frac{a(t)^T \bar{w} - c(t)}{\mu} - 1} d\lambda(t) = 0$. Hence, there exists a sufficiently small Θ_1 such that for $0 < \mu < \Theta_1$,

$$b^T \bar{w} - \mu \int_T e^{\frac{a(t)^T \bar{w} - c(t)}{\mu} - 1} d\lambda(t) > L.$$

This completes the proof. $\square$

Combining the lemmas derived before, we are now ready to prove the main result.

THEOREM 3.8. *Under Assumptions (A1)–(A3), if there exists $\hat{w} \in R^m$ such that $a(t)^T \hat{w} > 0$ a.e. in T , then there exists $\Theta > 0$ such that w_μ^* lies in a compact set $S(\bar{l}) \cap B$ for $0 < \mu \leq \Theta$.*

Proof. Suppose that the claim is not true. Then, Lemma 3.7 implies that there exists an infinite sequence $\{w_{\mu_i}^*\}$ whose elements are not in $S(\bar{l}) \cap B$ and $b^T w_{\mu_i}^* \geq L$ for each μ_i . Since

$$\int_T a_j(t) e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) = b_j, \quad j = 1, \dots, m,$$

we have

$$(3.17) \quad \int_T (a(t)^T \hat{w}) e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) = b^T \hat{w} < \infty.$$

By Lemma 3.6, for each i , we have $\lambda(w_{\mu_i}^*, \epsilon) \geq \delta > 0$. Thus

$$\begin{aligned} &\int_T (a(t)^T \hat{w}) e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) \\ &\geq \int_{\{t | a(t)^T w_{\mu_i}^* - c(t) \geq \epsilon\}} (a(t)^T \hat{w}) e^{\frac{\epsilon}{\mu_i} - 1} d\lambda(t) \\ (3.18) \quad &= e^{\frac{\epsilon}{\mu_i} - 1} \int_{\{t | a(t)^T w_{\mu_i}^* - c(t) \geq \epsilon\}} (a(t)^T \hat{w}) d\lambda(t) \end{aligned}$$

Since $a(t)^T \hat{w} > 0$ a.e. in T and $\lambda(w_{\mu_i}^*, \epsilon) \geq \delta > 0$, $\int_{\{t | a(t)^T w_{\mu_i}^* - c(t) \geq \epsilon\}} (a(t)^T \hat{w}) d\lambda(t)$ is bounded away from zero. Hence, the right-hand side of equation (3.18) approaches ∞ as $i \rightarrow \infty$. This contradicts equation (3.17) and completes the proof. $\square$

Consequently, the issues (ii) and (iii) have been taken care of by the following corollary.

COROLLARY 3.9. *Under Assumptions (A1)–(A3), if there exists $\hat{w} \in R^m$ such that $a(t)^T \hat{w} > 0$ a.e. in T , then there exists $\Theta > 0$ such that w_μ^* lies in a compact set $S(\bar{l}) \cap B$ for $0 < \mu \leq \Theta$. Moreover, any sequence $\{w_{\mu_i}^*\}$ with $\mu_i \rightarrow 0$ has at least one convergent subsequence in $S(\bar{l}) \cap B$, whose limiting point is an optimal solution of Program (D).*

4. Dual unboundedness and primal convergence in optimal value. Note that (A3) is an assumption on bounded level sets. To detect the unboundedness of the dual program (D), the following lemma may help.

LEMMA 4.1. *With Assumptions (A1) and (A2), suppose there exists $\hat{w} \in R^m$ such that $a(t)^T \hat{w} > 0$ a.e. in T . If $b^T w_{\mu_i}^*$ goes to infinity as μ_i decreases to 0, then either Program (D) is unbounded above or its optimal solution set is unbounded.*

Proof. Assume our conclusion is false, then since the feasible region of (D) is closed, Program (D) is bounded above and its optimal solution set is nonempty and bounded. In this case, we let L be the optimal objective value which is sufficient for Assumption (A3) to hold.

With Assumptions (A1) and (A2) and the existence of $\hat{w} \in R^m$ such that $a(t)^T \hat{w} > 0$ a.e. in T , Theorem 3.8 shows that the sequence $\{w_{\mu_i}^*\}$ lies in a compact set. This clearly causes a contradiction and the proof follows. $\square$

We now turn our attention to investigate the primal convergence of Program (P_{μ_i}) in terms of the optimal objective value. It is important to point out that in our formulation of Program (P), since the primal variable is

$$x(t) \in L_1^+(T, \lambda),$$

where $\lambda(t)$ is the Lebesgue measure, a particular regular Borel measure, Program (P) may not achieve P -attainment [2, 3, 4]. In other words, Program (P) may not have an optimal solution $x^*(t)$ such that

$$x^*(t) = \lim_{\mu_i \rightarrow 0} x_{\mu_i}^*(t) = \lim_{\mu_i \rightarrow 0} \int_T e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i}}^{-1} d\lambda(t)$$

and $x^*(t) \in L_1^+(T, \lambda)$. However, this does not affect the dual convergence results, and we shall show the convergence of Program (P_{μ_i}) in terms of optimal objective value.

We start with the following two lemmas.

LEMMA 4.2. *If $\{w_{\mu_i}^*\} \rightarrow w^*$ as $\mu_i \rightarrow 0$, then, for any $K > 0$, there exists $N > 0$ such that, for each $i > N$, $a(t)^T w_{\mu_i}^* - c(t) \leq K \forall t \in T$.*

Proof. If the claim is not true, then there exists a particular K such that, for any $N > 0$, there exists $i_N > N$ and $t_{i_N} \in T$ such that $a(t_{i_N})^T w_{\mu_{i_N}}^* - c(t_{i_N}) > K$.

Since $\{t_{i_N}\}$ lies in a compact set T , there is a subsequence which converges to $\bar{t}$ with

$$a(\bar{t})^T w^* - c(\bar{t}) \geq K.$$

This contradicts the fact that

$$a(t)^T w^* - c(t) \leq 0 \forall t \in T.$$

Hence, the proof is complete. $\square$

Note that when $w_{\mu_i}^*$ is optimal to Program (D_{μ_i}) , equation (2.4) holds as

$$(4.1) \quad \int_T a_j(t) e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) = b_j$$

for $j = 1, \dots, m$. Now, if there exists $\hat{w}$ and $M > 0$ such that $a(t)^T \hat{w} \geq M \quad \forall t \in T$, then equation (4.1) implies that

$$\int_T (a(t)^T \hat{w}) e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) = b^T \hat{w}$$

and

$$(4.2) \quad 0 \leq \int_T e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) \leq \frac{b^T \hat{w}}{M}.$$

This leads to the following result.

LEMMA 4.3. *If there exists $\hat{w} \in R^m$ and $M > 0$ such that $a(t)^T \hat{w} \geq M \quad \forall t \in T$, then $b^T \hat{w} \geq 0$ and*

$$0 \leq \int_T e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) \leq \frac{b^T \hat{w}}{M} \quad \forall i.$$

Combining Lemmas 4.2 and 4.3, we have the following convergence result.

THEOREM 4.4. *If $\{w_{\mu_i}^*\} \rightarrow w^*$ as $\mu_i \rightarrow 0$ and there exists $\hat{w} \in R^m$ and $M > 0$ such that $a(t)^T \hat{w} \geq M > 0 \quad \forall t \in T$, then*

$$\lim_{\mu_i \rightarrow 0} \mu_i \int_T x_{\mu_i}^*(t) \log x_{\mu_i}^*(t) d\lambda(t) = 0$$

and

$$\lim_{\mu_i \rightarrow 0} \int_T c(t) x_{\mu_i}^*(t) d\lambda(t) = b^T w^*.$$

Proof. Given $\epsilon > 0$, let $K = M\epsilon/b^T \hat{w}$. Lemma 4.2 provides an $N > 0$ such that, for $i > N$,

$$(4.3) \quad a(t)^T w_{\mu_i}^* - c(t) \leq \frac{M\epsilon}{b^T \hat{w}} \quad \forall t \in T.$$

Then from equations (2.3), (4.2), and (4.3), for $i > N$, we have

$$\begin{aligned} & \mu_i \int_T x_{\mu_i}^*(t) \log x_{\mu_i}^*(t) d\lambda(t) \\ &= \int_T (a(t)^T w_{\mu_i}^* - c(t)) e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) \\ &\leq \int_T \frac{M\epsilon}{b^T \hat{w}} e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) \\ &\leq \frac{M\epsilon}{b^T \hat{w}} \cdot \frac{b^T \hat{w}}{M} = \epsilon. \end{aligned}$$

Therefore,

$$\lim_{\mu_i \rightarrow 0} \mu_i \int_T x_{\mu_i}^*(t) \log x_{\mu_i}^*(t) d\lambda(t) = 0.$$

Since

$$\int_T c(t) x_{\mu_i}^*(t) d\lambda(t) + \mu_i \int_T x_{\mu_i}^*(t) \log x_{\mu_i}^*(t) d\lambda(t) = b^T w_{\mu_i}^* - \mu_i \int_T e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t)$$

as $\mu_i \rightarrow 0$, we have

$$\lim_{\mu_i \rightarrow 0} \int_T c(t) x_{\mu_i}^*(t) d\lambda(t) = b^T w^*. \quad \square$$

This clearly shows the primal convergence of Program (P_{μ_i}) in terms of the optimal objective value.

5. Numerical examples. In this section, three numerical examples are reported. Our purpose is not to claim any computational superiority of the proposed method. Instead, we simply intend to illustrate the computational behavior of the proposed approach. Note that the proposed approach is flexible enough to use any commercial or public unconstrained nonlinear optimizer, instead of developing special codes for our own implementation. In our examples, the L-BFGS-B software [21] was applied.

5.1. L1 problems. In this subsection, the following two commonly seen L1 problems [14, 15] were tested.

Problem 1.

$$\begin{aligned} & \max \sum_{j=1}^8 \frac{-w_j}{j} \\ \text{s.t. } & \sum_{j=1}^8 -t^{j-1} w_j \leq \frac{-1}{2-t}, \quad t \in [0, 1]. \end{aligned}$$

Problem 2.

$$\begin{aligned} & \max \sum_{j=1}^7 \frac{-w_j}{j} \\ \text{s.t. } & \sum_{j=1}^7 -t^{j-1} w_j \leq \sum_{j=0}^4 t^{2j}, \quad t \in [0, 1]. \end{aligned}$$

It was reported in [14, 15] that 0.6931 and -1.78688 are approximate optimal solutions to Problems 1 and 2, respectively.

We applied the L-BFGS-B subroutines [21] to solve Program (D_{μ}) with the standard default setting in “driver1.f” of the L-BFGS-B solver. The stopping criteria of L-BFGS-B was given by

$$\frac{(f_k - f_{k+1})}{\max(|f_k|, |f_{k+1}|, 1)} \leq \text{factr} * \text{epsmch},$$

TABLE 5.1
 Problem 1.

μ_i	$v(D_{\mu_i})$	$b^T w_{\mu_i}^*$	argmin_t	min const.	comp. slackness
0.0156250	0.6931472	0.6775222	1.0000000	0.0156904	0.0156243
0.0078125	0.6931472	0.6853348	0.0000025	0.0078887	0.0078053
0.0039062	0.6931472	0.6892410	0.9999900	0.0039880	0.0038958
0.0019531	0.6931475	0.6911944	1.0000000	0.0020775	0.0019488
0.0009766	0.6931472	0.6921706	0.9999875	0.0010074	0.0009626
0.0004883	0.6931476	0.6926593	0.9999950	0.0005731	0.0005987
0.0002441	0.6931482	0.6929041	0.9999975	0.0003303	0.0002555
0.0001221	0.6931473	0.6930252	0.9999850	0.0001465	0.0002011
0.0000610	0.6931477	0.6930866	0.9999950	0.0000913	0.0001282

 TABLE 5.2
 Problem 2.

μ_i	$v(D_{\mu_i})$	$b^T w_{\mu_i}^*$	argmin_t	min const.	comp. slackness
0.0156250	-1.7872568	-1.8028820	0.9225800	0.0175587	0.0156645
0.0078125	-1.7872132	-1.7950259	0.9219850	0.0096138	0.0078382
0.0039062	-1.7871293	-1.7910354	0.9200125	0.0055678	0.0041313
0.0019531	-1.7869809	-1.7889341	0.9160275	0.0033668	0.0022031
0.0009766	-1.7867584	-1.7877351	0.9093575	0.0020499	0.0010036
0.0004883	-1.7865123	-1.7870006	0.9025550	0.0012140	0.0006634
0.0002441	-1.7862819	-1.7865261	0.8980550	0.0006985	0.0003202
0.0001221	-1.7861237	-1.7862457	0.8967650	0.0003929	0.0005766
0.0000610	-1.7860207	-1.7860818	0.8956500	0.0002182	-0.0008398

where f is the function to be optimized and f^k, f^{k+1} are the values of f in the k th and $(k+1)$ th iteration, respectively. Moreover, $epsmch = 2.22 \times 10^{-16}$, which reflects the machine accuracy for our Silicon Graphics workstation, and $factr$ was set to be 10^5 .

A trivial initial solution with $w_j = 1, j = 1, \dots, 8$ was chosen for Problem 1. The numerical results of the proposed approach are shown in Table 5.1.

In the table, “ argmin_t ” = $\text{argmin}\{a(t)^T w_{\mu_i}^* - c(t)\}$, “min const.” = $\min_t\{a(t)^T w_{\mu_i}^* - c(t)\}$, and “comp. slackness” =

$$\begin{aligned}
 & \int_T c(t) e^{\frac{a(t)^T w_{\mu_i}^* - c(t)}{\mu_i} - 1} d\lambda(t) - b^T w_{\mu_i}^* \\
 &= \int_T c(t) x_{\mu_i}^*(t) d\lambda(t) - b^T w_{\mu_i}^* \\
 &= -\mu_i \int_T x_{\mu_i}^*(t) \log x_{\mu_i}^*(t) d\lambda(t) - \mu_i \int_T x_{\mu_i}^*(t) d\lambda(t).
 \end{aligned}$$

The final $w_{\mu_i}^*$ we obtained for Problem 1 is

$$[0.4999164, 0.2507643, 0.1192728, 0.0773788, 0.0201207, 0.0092012, 0.0123938, 0.0108609]^T.$$

Similarly, we chose a trivial initial solution with $w_j = 0, j = 1, \dots, 7$ for Problem 2. The numerical results are shown in Table 5.2.

The final $w_{\mu_i}^*$ we obtained for Problem 2 is

$$[-0.9927071, -0.1584586, 0.1286875, -3.0721959, 1.5562050, 2.0875959, -4.5371615]^T.$$

TABLE 5.3
Two-dimensional problem.

μ_i	$v(D_{\mu_i})$	$b^T w_{\mu_i}^*$	argmin_s	argmin_t	min const.	comp. slackness
0.0009766	2.4276713	2.4266948	1.0000000	1.0000000	0.0172085	0.0073562
0.0004883	2.4310292	2.4305409	1.0000000	1.0000000	0.0105747	0.0043555
0.0002441	2.4328797	2.4326356	1.0000000	1.0000000	0.0072397	0.0021812
0.0001221	2.4338841	2.4337620	1.0000000	1.0000000	0.0055717	0.0011672
0.0000610	2.4344216	2.4343605	1.0000000	1.0000000	0.0047382	0.0003050
0.0000305	2.4347056	2.4346750	1.0000000	1.0000000	0.0043210	-0.0002368
0.0000153	2.4348551	2.4348398	1.0000000	1.0000000	0.0041131	-0.0015739
0.0000076	2.4349335	2.4349259	1.0000000	1.0000000	0.0040091	-0.0079867
0.0000038	2.4349818	2.4349780	1.0000000	1.0000000	0.0039562	-0.0010248
0.0000019	2.4349964	2.4349945	1.0000000	1.0000000	0.0039319	0.0048256
0.0000010	2.4350420	2.4350411	1.0000000	1.0000000	0.0039185	0.0000394

The first three columns of both tables clearly show that $b^T w_{\mu_i}^*$ converges to the reported solution up to the 4th digit after the decimal point as μ_i decreases. The last two columns of both tables also show that the dual feasibility and complementary slackness conditions are satisfied up to the 4th digit after the decimal point, when μ_i is small enough.

The computational bottleneck of the proposed approach lies in the following integration:

$$(5.1) \quad \int_T e^{\frac{a(t)^T w - c(t)}{\mu} - 1} d\lambda(t).$$

Numerically, we need to discretize T for integration. When μ_i is getting smaller, a finer discretization of T is needed. This could be very time consuming, especially when T has a high dimensionality. In our examples, we partition $T = [0, 1]$ into 400,000 intervals and use the Simpson's method for integration.

5.2. Two-dimensional problem. In this subsection, the following two-dimensional problem on page 112 of [15] was tested:

$$\begin{aligned} \min w_1 + \frac{w_2}{2} + \frac{w_3}{2} + \frac{w_4}{4} + \frac{w_5}{4} + \frac{w_6}{3} \\ \text{s.t. } w_1 + t_1 w_2 + t_2 w_3 + t_1^2 w_4 + t_1 t_2 w_5 + t_2^2 w_6 \geq -e^{t_1^2 + t_2^2}, \quad t_1 \in [0, 1], t_2 \in [0, 1] \\ |w_j| \leq 10, \quad j = 1, \dots, 6. \end{aligned}$$

It was reported in [7] that 2.4356 is an approximate optimal objective value. We followed all settings as described in the previous subsection with an initial solution $w_1 = 7.4$ and $w_i = 0$ for $i = 2, \dots, 6$. The numerical results are shown in Table 5.3.

The final $w_{\mu_i}^*$ we obtained is

$$[2.5759610, -4.0618098, -4.1392911, 4.2036264, 4.5255433, 4.2811078]^T.$$

For this two-dimensional problem, we partitioned $[0, 1] \times [0, 1]$ into 1500^2 intervals for integration. The computational behavior of this case is quite similar to those reported in the previous subsection. The proposed approach indeed generates a convergent solution which satisfies the dual feasibility and complementary slackness conditions.

Because the numerical integration of $\int_T e^{\frac{a(t)^T w - c(t)}{\mu} - 1} d\lambda(t)$ could be difficult for a general problem, we do not claim any computational superiority of the proposed

approach. However, the proposed approach does provide a new angle to solve the linear semi-infinite programming problems.

6. Concluding remarks.

1. We have proposed an unconstrained convex programming approach to solving linear semi-infinite programming problems. In this approach, solving Program (D_μ) with a sufficiently small $\mu > 0$ provides an approximate solution to Program (D) .
2. A dual convergence result shows that, under Assumptions (A1)–(A3), $(D_\mu) \rightarrow (D)$ as $\mu \rightarrow 0$, in terms of optimal solutions.
3. A primal convergence result shows that $(P_\mu) \rightarrow (P)$ as $\mu \rightarrow 0$, in terms of optimal objective values.
4. Compared to most known methods, the proposed approach does not require any search procedures for finding a global optimizer over T . Instead, an integration over T is required. We do not claim any computational superiority of the proposed approach. However, it does provide an alternative. For some problems, it might be easier to do integration than to perform a global optimization over T .
5. The proposed approach is flexible to use any unconstrained nonlinear optimizer. Three examples were included to illustrate the proposed approach.

Acknowledgments. The authors would like to thank Dr. A. R. Conn and referees for their constructive comments.

REFERENCES

- [1] M. AVRIEL, *Nonlinear Programming: Analysis and Methods*, Prentice-Hall, Englewood Cliffs, NJ, 1976.
- [2] J. M. BORWEIN AND A. S. LEWIS, *Convergence of best entropy estimates*, SIAM J. Optim., 1 (1991), pp. 191–205.
- [3] J. M. BORWEIN AND A. S. LEWIS, *Partially-finite programming in L_1 and the existence of maximum entropy estimates*, SIAM J. Optim., 3 (1993), pp. 248–267.
- [4] J. M. BORWEIN AND A. S. LEWIS, *A survey of convergence results for maximum entropy methods*, in Maximum Entropy and Bayesian Methods, A. Mohammad-Djafari and G. Demoments, eds., Kluwer Academic Publishers, Norwell, MA, 1993, pp. 39–48.
- [5] D. BOUKARI AND A. V. FIACCO, *Survey of penalty, exact-penalty and multiplier methods from 1968 to 1993*, Optimization, 32 (1995), pp. 301–334.
- [6] A. R. CONN AND N. I. M. COULD, *An exact penalty function for semi-infinite programming*, Math. Programming, 37 (1987), pp. 19–40.
- [7] I. D. COOPE AND G. A. WATSON, *A projected Lagrangian algorithm for semi-infinite programming*, Math. Programming, 32 (1985), pp. 337–356.
- [8] J. E. DENNIS AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1983.
- [9] J. P. EVANS AND F. J. GOULD, *An existence theorem for penalty function theory*, SIAM J. Control Optim., 12 (1974), pp. 509–516.
- [10] S.-C. FANG AND S. PUTHENPURA, *Linear Optimization and Extensions: Theory and Algorithms*, Prentice-Hall, Englewood Cliffs, NJ, 1993.
- [11] S.-C. FANG AND H.-S. J. TSAO, *Linear programming with entropic perturbation*, Z. Oper. Res., 37 (1993), pp. 171–186.
- [12] S.-C. FANG AND H.-S. J. TSAO, *On the entropic perturbation and exponential penalty methods for linear programming*, J. Optim. Theory Appl., 89 (1996), pp. 461–466.
- [13] S.-C. FANG AND S.-Y. WU, *An inexact approach to solving linear semi-infinite programming problems*, Optimization, 28 (1994), pp. 291–299.
- [14] M. C. FERRIS AND A. B. PHILPOTT, *An interior point algorithm for semi-infinite linear programming*, Math. Programming, 43 (1989), pp. 257–276.
- [15] K. GLASHOFF AND S. Å. GUSTAFSON, *Linear Optimization and Approximation*, Springer-Verlag, New York, 1982.

- [16] S. Å. GUSTAFSON, *On the computational solution of a class of generalized moment problems*, SIAM J. Numer. Anal., 7 (1970), pp. 343–357.
- [17] R. HETTICH AND K. KORTANEK, *Semi-infinite programming: Theory, method, and applications*, SIAM Rev., 35 (1993), pp. 380–429.
- [18] C.-J. LIN, S.-C. FANG, AND S.-Y. WU, *A dual affine scaling based algorithm for solving linear semi-infinite programming problems*, in *Advances in Optimization and Approximation*, Ding-Zhu Du and Jie Sun, eds., Kluwer Academic Publishers, Norwell, MA, 1994, pp. 217–234.
- [19] T. PIETRZYKOWSKI, *The potential method for conditional maxima in the locally compact metric spaces*, Numer. Math., 14 (1970), pp. 325–329.
- [20] R. T. ROCKFALLER, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
- [21] C. ZHU, R. H. BYRD, P. LU, AND J. NOCEDAL, *L-BFGS-B Fortran subroutines for large-scale bound constrained optimization*, Tech. report, Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, 1994.

THE LINEAR ℓ_1 ESTIMATOR AND THE HUBER M-ESTIMATOR*

W. LI[†] AND J. J. SWETITS[†]

Abstract. Relationships between a linear ℓ_1 estimation problem and the Huber M-estimator problem can be easily established by their dual formulations. The least norm solution of a linear programming problem studied by Mangasarian and Meyer [*SIAM J. Control Optim.*, 17 (1979), pp. 745–752] provides a key link between the dual problems. Based on the dual formulations, we establish a local linearity property of the Huber M-estimators with respect to the tuning parameter γ and prove that the solution set of the Huber M-estimator problem is Lipschitz continuous with respect to perturbations of the tuning parameter γ . As a consequence, the set of the linear ℓ_1 estimators is the limit of the set of the Huber M-estimators as $\gamma \rightarrow 0^+$. Thus, the Huber M-estimator problem has many solutions for small tuning parameter γ if the linear ℓ_1 estimation problem has multiple solutions. A recursive version of Madsen and Nielsen’s algorithm [*SIAM J. Optim.*, 3 (1993), pp. 223–235] based on computation of the Huber M-estimator is proposed for finding a linear ℓ_1 estimator.

Key words. linear ℓ_1 estimator, Huber M-estimator, least norm solution of linear programs, dual solutions of least distance problems

AMS subject classifications. Primary, 90C90, 62G35; Secondary, 41A45, 41A52, 90C31

PII. S1052623495293160

1. Introduction. Consider the following linear ℓ_1 estimation problem of finding a vector x^* in $\mathbf{R}^n$ that solves the following minimization problem:

$$(1) \quad \min_x \|Ax - d\|_1,$$

where A is an $m \times n$ ($m > n$) matrix representing the underlying linear model, x in $\mathbf{R}^n$ is the parameter vector, $d \in \mathbf{R}^m$ is the data vector, and $\|z\|_1 := \sum_{i=1}^m |z_i|$ is the ℓ_1 -norm of z . In general, one might assume that A has rank n so that different parameter vectors correspond to different regression functions, even though this assumption is not required in our study of (1).

The linear ℓ_1 estimation problem is much more difficult to solve than the linear maximum likelihood estimation problem, where the ℓ_2 -norm, $\|z\|_2 := (\sum_{i=1}^m z_i^2)^{\frac{1}{2}}$, is used instead of the ℓ_1 -norm. See [1, 2, 3, 9, 19, 26, 28, 30, 31] for various algorithms for solving the linear ℓ_1 estimation problem. One reason for solving (1) instead of the corresponding least squares problem is that linear ℓ_1 estimators are more robust than linear least squares estimators [11, 12] in the presence of outliers. Note that (1) is referred to as a nonsmooth optimization problem because the objective function (i.e., the expression one tries to minimize) is not a differentiable function of parameters $x_1, \dots, x_n$. Many robust regression models allow one to get a robust estimator without involving nonsmooth optimization problems [12]. The most well-known one is the Huber M-estimator problem that combines the robustness of the linear ℓ_1 regression

*Received by the editors October 13, 1995; accepted for publication (in revised form) March 4, 1997. This research was partially supported by NASA/Langley Research Center under grant NCC-1-68 supplement-16. The research of the first author was partially supported by the AFOSR under grant 94-NM-390.

<http://www.siam.org/journals/siopt/8-2/29316.html>

[†]Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529 (wuli@math.odu.edu).

model with the smoothness of the linear maximum likelihood regression model by using Huber's cost function, depending on a given tuning constant $\gamma > 0$, defined by

$$\rho(t) = \begin{cases} \frac{1}{2}t^2, & \text{if } |t| \leq \gamma, \\ \gamma|t| - \frac{1}{2}\gamma^2, & \text{otherwise.} \end{cases}$$

The Huber M-estimator problem is to find a vector x^* in $\mathbf{R}^n$ such that x^* solves the following minimization problem:

$$(2) \quad \min_x \sum_{i=1}^m \rho[(Ax - d)_i],$$

where $(Ax - d)_i$ denotes the i th component of the vector. In general, there is a scaling parameter τ involved in the Huber M-estimator problem. For convenience, we assume that $\tau \equiv 1$. Note that Huber's cost function is a convex differentiable piecewise quadratic polynomial and the objective function in (2) is differentiable. The purpose of Huber's cost function is to measure $(Ax - d)_i$ by $(Ax - d)_i^2$ (as in maximum likelihood estimation) when $(Ax - d)_i$ is "relatively small" while it measures $(Ax - d)_i$ by $|(Ax - d)_i|$ (as in linear ℓ_1 estimation) when d_i is an outlier (i.e., $|(Ax - d)_i|$ is "relatively large").

Recently, there have been many papers devoted to design of numerical algorithms for solving the Huber M-estimator problem [6, 7, 15, 18, 24, 29], as well as to relationships between linear ℓ_1 estimators and Huber M-estimators [5, 19, 20]. Madsen and Nielsen [19] and Madsen, Nielsen, and Pinar [20] show that algorithms for computing Huber M-estimators can also be used to find linear ℓ_1 estimators by establishing an explicit correspondence between linear ℓ_1 estimators and Huber M-estimators. The purpose of this paper is to show that more revealing relationships between linear ℓ_1 estimators and Huber M-estimators can be easily derived through dual formulations of (1) and (2). More specifically, we prove that the dual solution of the Huber M-estimator problem is the least norm solution of the dual linear programming problem of (1) when the tuning parameter $\gamma > 0$ is small enough. The proof is based on a characterization of the least norm solution of a linear program proved by Mangasarian [21] and Mangasarian and Meyer [22]. As consequences, we not only recover the known results about linear ℓ_1 estimators and Huber M-estimators, but also derive new ones. This allows us to explicitly verify whether a given parameter γ is small enough to produce a linear ℓ_1 estimator. We will propose a recursive version of Madsen and Nielsen's algorithm for finding a linear ℓ_1 estimator by solving the corresponding Huber M-estimator problem.

It is well known that either (1) or (2) may have infinitely many solutions. Sufficient conditions for uniqueness of the linear ℓ_1 estimator and the Huber M-estimator are given in [5, 18, 19, 20]. However, it is not clear whether there is a connection between the uniqueness of the linear ℓ_1 estimator and the Huber M-estimator. We prove that the uniqueness of the Huber M-estimator for every small tuning factor γ implies the uniqueness of the linear ℓ_1 estimator. However, a counterexample shows that the converse does not hold. That is, there exist A and d such that (1) has a unique solution, but the corresponding Huber M-estimator problem (2) has infinitely many solutions for every $0 < \gamma \leq \delta$, where δ is a positive constant.

The paper is organized as follows. Dual problems of (1) and (2) are given in section 2 and characterizations of solutions of (1) and (2) are also given in terms of their dual solutions, respectively. In section 3, we first prove the Lipschitz stability

of the solution set X^γ of (2) with respect to the perturbations of γ . Then, based on the dual characterizations given in the previous section, we establish a local linearity property of X^γ as a set-valued mapping of the parameter γ when $\gamma > 0$ is small enough. Moreover, we investigate the relationship between uniqueness of the Huber M-estimator and uniqueness of the linear ℓ_1 -estimator. In section 4, a recursive algorithm for finding a linear ℓ_1 estimator is proposed. Conclusions are included in section 5.

To conclude the section we give some notations used in this paper. If A is a matrix and x is a (column) vector, A_i denotes the i th row of A and x_i denotes the i th component of x . (Note $A_i x \equiv (Ax)_i$.) For a vector x (or a matrix A), x^T (or A^T) is its transpose. For two vectors x and y , $x \leq y$ means $x_i \leq y_i$ for every index i . For convenience, when x is a vector and α is a number, $x \leq \alpha$ denotes $x_i \leq \alpha$ for every index i . We use $z_{-\gamma}^\gamma$ to denote the vector whose i th component is $\max\{-\gamma, \min\{\gamma, z_i\}\}$. The ℓ_∞ -norm, $\|\cdot\|_\infty$, of a vector $z \in \mathbf{R}^m$ is $\|z\|_\infty = \max_{1 \leq i \leq m} |z_i|$. The set of all solutions of (1) is denoted by X^0 , while the set of all solutions of (2) is denoted by X^γ for any given tuning parameter γ . The null space of a matrix A is defined as $\{x \in \mathbf{R}^n : Ax = 0\}$.

2. Dual formulations. In this section we give the dual formulations of (1) and (2) as a linear programming problem and a quadratic programming problem, respectively. These dual formulations provide characterizations of linear ℓ_1 estimators and Huber M-estimators in terms of their dual solutions. Then, from Mangasarian and Meyer's characterization of the least norm solution of a linear program [22, 21], we know that the unique dual solution of (2) is actually the least norm dual solution of (1) when $\gamma > 0$ is small enough.

Consider the following dual formulation of (1):

$$(3) \quad \min_y d^T y \quad \text{subject to } A^T y = 0, \quad -1 \leq y \leq 1.$$

The following lemma is essentially known, but the form of the statement given here is more convenient for our purposes. The proof is included for the sake of completeness. For another form of the lemma, see [25, 20].

LEMMA 2.1. *Let $\bar{x} \in \mathbf{R}^n$ and $\bar{y} \in \mathbf{R}^m$. Then $\bar{x}$ and $\bar{y}$ are solutions of (1) and (3), respectively, if and only if the following conditions hold:*

$$(4) \quad \begin{aligned} A^T \bar{y} &= 0, \quad -1 \leq \bar{y} \leq 1, \\ (A\bar{x} - d)_i &\geq 0, \quad \text{if } \bar{y}_i = 1, \\ (A\bar{x} - d)_i &\leq 0, \quad \text{if } \bar{y}_i = -1, \\ (A\bar{x} - d)_i &= 0, \quad \text{if } -1 < \bar{y}_i < 1. \end{aligned}$$

Proof. First suppose that (4) holds. Let $x \in \mathbf{R}^n$ and $y \in \mathbf{R}^m$ with $A^T y = 0$, $-1 \leq y \leq 1$. Then

$$(5) \quad \begin{aligned} \|Ax - d\|_1 &\geq \bar{y}^T (Ax - d) = -\bar{y}^T d = \bar{y}^T (A\bar{x} - d) \\ &= \|A\bar{x} - d\|_1 \geq y^T (A\bar{x} - d) = -y^T d, \end{aligned}$$

where the inequalities follow from $-1 \leq y, \bar{y} \leq 1$ and the equalities follow from $A^T y = A^T \bar{y} = 0$. Therefore, $\|A\bar{x} - d\|_1 = \min_x \|Ax - d\|_1$ and $d^T y \geq d^T \bar{y}$ for $A^T y = 0$, $-1 \leq y \leq 1$. Thus, $\bar{x}$ and $\bar{y}$ are solutions of (1) and (3), respectively.

Now suppose that $\bar{x}$ and $\bar{y}$ are solutions of (1) and (3), respectively. By the KKT conditions for $\bar{y}$, there exist $x^* \in \mathbf{R}^n$ and $z^* \in \mathbf{R}^m$ such that

$$\begin{aligned}
d - Ax^* - z^* &= 0, \\
A^T \bar{y} &= 0, \quad -1 \leq \bar{y} \leq 1, \\
z_i^* &\leq 0, \text{ if } \bar{y}_i = 1, \\
z_i^* &\geq 0, \text{ if } \bar{y}_i = -1, \\
z_i^* &= 0, \text{ if } -1 < \bar{y}_i < 1.
\end{aligned}$$

Substituting $z^* = d - Ax^*$ into the last three inequalities/equalities we obtain

$$\begin{aligned}
(Ax^* - d)_i &\geq 0, \text{ if } \bar{y}_i = 1, \\
(Ax^* - d)_i &\leq 0, \text{ if } \bar{y}_i = -1, \\
(Ax^* - d)_i &= 0, \text{ if } -1 < \bar{y}_i < 1.
\end{aligned} \tag{6}$$

It follows from (6) that $\|Ax^* - d\|_1 = \bar{y}^T(Ax^* - d)$. Then we can derive that

$$\|Ax^* - d\|_1 = \bar{y}^T(Ax^* - d) = \bar{y}^T(A\bar{x} - d) \leq \|A\bar{x} - d\|_1 \leq \|Ax^* - d\|_1. \tag{7}$$

Here the first equality is the result of (6) and the second equality follows from $A^T \bar{y} = 0$; the first inequality is derived from $-1 \leq \bar{y} \leq 1$; and since $\bar{x}$ is a solution of (1), we have the last inequality. Therefore, all equalities in (7) must hold and

$$\bar{y}^T(A\bar{x} - d) = \|A\bar{x} - d\|_1. \tag{8}$$

Since $-1 \leq \bar{y} \leq 1$, equation (8) is equivalent to the last three inequalities/equalities in (4). $\square$

Note that the standard KKT conditions for (3) involve a multiplier $\bar{z}$ for the constraints $-1 \leq y \leq 1$ and an equation $d - A\bar{x} - \bar{z} = 0$. Here we use $\bar{z} := -(A\bar{x} - d)$ to eliminate $\bar{z}$ in the above lemma. The following characterization of linear ℓ_1 estimators based on a solution to (3) follows immediately from Lemma 2.1.

COROLLARY 2.2. *Let y^* be a solution of (3). Then $\bar{x}$ is a solution of (1) if and only if $\bar{x}$ satisfies the following conditions:*

$$\begin{aligned}
(A\bar{x} - d)_i &\geq 0, \text{ if } y_i^* = 1, \\
(A\bar{x} - d)_i &\leq 0, \text{ if } y_i^* = -1, \\
(A\bar{x} - d)_i &= 0, \text{ if } -1 < y_i^* < 1.
\end{aligned} \tag{9}$$

The classical approximation theoretic characterization of solutions to (1) states that $\bar{x} \in \mathbf{R}^n$ solves (1) if and only if there exists $\bar{y} \in \mathbf{R}^m$ such that $-1 \leq \bar{y} \leq 1$, $A^T \bar{y} = 0$ and $\bar{y}^T(A\bar{x} - d) = \|A\bar{x} - d\|_1$ (cf. [30, Chapter 6]). It is easy to see that these conditions are equivalent to (4) or (9).

Now consider the least norm solution of (3), which coincides, when $\gamma > 0$ is sufficiently small, with the unique solution of the following quadratic program:

$$\min_y d^T y + \frac{\gamma}{2} \|y\|_2^2 \quad \text{subject to } A^T y = 0, -1 \leq y \leq 1, \tag{10}$$

as shown in the following lemma by Mangasarian and Meyer [22, 21].

LEMMA 2.3. *Let y^γ be the unique solution of the strictly convex quadratic program (10). Then there exists $\delta > 0$ such that y^γ is the least norm solution of (3) for $0 < \gamma < \delta$.*

Remark. This lemma can also be obtained by using weak-sharp minima of linear programs [8, 4].

Here we say that y^* is the least norm solution of (3) if y^* is the solution of the following strictly convex quadratic programming problem:

$$\min_{y \in Y^*} \frac{1}{2} \|y\|_2^2,$$

where Y^* is the set of all solutions to (3). That is, y^* is the least norm solution of (3) if y^* is the solution of (3) that has the smallest ℓ_2 -norm. The least norm solution of (3) is actually the unique common solution of (3) and (10).

LEMMA 2.4. *If the solution y^* of (10) is also a solution of (3), then y^* is the least norm solution of (3).*

Proof. Let $\bar{y}$ be any solution of (3). Since y^* is a solution of (10) and $A^T \bar{y} = 0$, $-1 \leq \bar{y} \leq 1$, we obtain that

$$(11) \quad d^T y^* + \frac{\gamma}{2} \|y^*\|_2^2 \leq d^T \bar{y} + \frac{\gamma}{2} \|\bar{y}\|_2^2.$$

Since y^* and $\bar{y}$ are solutions of (3), $d^T y^* = d^T \bar{y}$. Thus, (11) implies that $\|y^*\|_2^2 \leq \|\bar{y}\|_2^2$. So y^* is the least norm solution of (3). $\square$

Since $\rho'(t) = (t)_{-\gamma}^\gamma$, the gradient of the objective function in (2) is $A^T(Ax - d)_{-\gamma}^\gamma$. Since (2) is a convex differentiable optimization problem, x^* is a solution of (2) if and only if x^* is a zero of the gradient of the objective function; i.e., x^* is a solution of the following system of piecewise linear equations:

$$(12) \quad A^T(Ax - d)_{-\gamma}^\gamma = 0.$$

For easy reference, we give the following lemma for the equivalence of (12) and (2).

LEMMA 2.5. *A vector x^γ is a solution of (12) if and only if x^γ is a solution of (2).*

It is known that (12) can be considered as a dual problem of (10) [17, 24]. Therefore, (2) and (10) are dual problems. Here is the relation between solutions of (10) and (2) [17, 24, 13].

LEMMA 2.6. *A vector y^γ is the solution of (10) if and only if $y^\gamma = \frac{1}{\gamma}(Ax^\gamma - d)_{-\gamma}^\gamma$, where x^γ is a solution of (2).*

Therefore, in order to get the (unique) solution of (10), one only needs to find a Huber M-estimator by solving (2) (or (12)). Many algorithms (including Newton methods, matrix splitting methods, and conjugate gradient methods) were developed for solving the quadratic program (10) by exploring its dual structure (12) [17, 24, 13].

For a better understanding of the duality between (2) and (10), we recast Lemma 2.6 as follows.

LEMMA 2.7. *Let $\bar{x} \in \mathbf{R}^n$ and $\bar{y} \in \mathbf{R}^m$. Then $\bar{x}$ and $\bar{y}$ are solutions of (2) and (10), respectively, if and only if the following conditions hold:*

$$(13) \quad \begin{aligned} A^T \bar{y} &= 0, \quad -1 \leq \bar{y} \leq 1, \\ (A\bar{x} - d)_i &\geq \gamma, \quad \text{if } \bar{y}_i = 1, \\ (A\bar{x} - d)_i &\leq -\gamma, \quad \text{if } \bar{y}_i = -1, \\ (A\bar{x} - d)_i &= \gamma \bar{y}_i, \quad \text{if } -1 < \bar{y}_i < 1. \end{aligned}$$

Proof. Suppose that (13) holds. Then we easily deduce that $\bar{y} = \frac{1}{\gamma}(A\bar{x} - d)_{-\gamma}^\gamma$ and $A^T(A\bar{x} - d)_{-\gamma}^\gamma = \gamma A^T \bar{y} = 0$. By Lemma 2.5, $\bar{x}$ is a solution of (2). By Lemma 2.6, $\bar{y}$ is a solution of (10).

Now suppose that $\bar{x}$ and $\bar{y}$ are solutions of (2) and (10), respectively. By Lemma 2.6, $\bar{y} = \frac{1}{\gamma}(A\bar{x} - d)_{-\gamma}^\gamma$, which implies the last three inequalities/equalities in (13). Since $\bar{y}$ is a solution of (10), we have $A^T \bar{y} = 0$, $-1 \leq \bar{y} \leq 1$. Thus, (13) holds. $\square$

Note that the standard KKT conditions for (10) involve a multiplier $\bar{z}$ for the constraints $-1 \leq y \leq 1$ and an equation $d - \gamma \bar{y} - A\bar{x} - \bar{z} = 0$. Here we use $\bar{z} := -(A\bar{x} - d - \gamma \bar{y})$ to eliminate $\bar{z}$ in the above lemma. The following characterization of the Huber M-estimators based on a solution to (3) follows immediately from Lemma 2.7.

COROLLARY 2.8. *Let y^* be a solution of (10). Then $\bar{x}$ is a solution of (2) if and only if $\bar{x}$ satisfies the following conditions:*

$$(14) \quad \begin{aligned} (A\bar{x} - d)_i &\geq \gamma, \text{ if } y_i^* = 1, \\ (A\bar{x} - d)_i &\leq -\gamma, \text{ if } y_i^* = -1, \\ (A\bar{x} - d)_i &= \gamma y_i^*, \text{ if } -1 < y_i^* < 1. \end{aligned}$$

Lemma 2.4 provides a way to verify whether or not the solution of (10) is the least norm solution of (3). By Lemma 2.3, there exists a positive constant δ such that the solution of (10) is the least norm solution y^* of (3) for $0 < \gamma \leq \delta$. But it does not tell us how small δ should be. The following lemma tells us that if the solution of (10) is the least norm solution y^* for some $\gamma = \delta > 0$, then y^* is also the solution of (10) for any $0 < \gamma < \delta$.

LEMMA 2.9. *If the solution y^δ of (10) is the least norm solution y^* of (3) for some $\gamma = \delta > 0$, then the solution y^γ of (10) is the least norm solution of (3) for any $0 < \gamma \leq \delta$ (i.e., $y^\gamma \equiv y^*$ for $0 < \gamma \leq \delta$).*

Proof. From Corollary 2.8 we have that $x^\delta \in X^\delta$ if and only if

$$(15) \quad \begin{aligned} (Ax^\delta - d)_i &\geq \delta, \text{ if } y_i^* = 1, \\ (Ax^\delta - d)_i &\leq -\delta, \text{ if } y_i^* = -1, \\ (Ax^\delta - d)_i &= \delta y_i^*, \text{ if } -1 < y_i^* < 1. \end{aligned}$$

By Corollary 2.2, $x^0 \in X^0$ if and only if

$$(16) \quad \begin{aligned} (Ax^0 - d)_i &\geq 0, \text{ if } y_i^* = 1, \\ (Ax^0 - d)_i &\leq 0, \text{ if } y_i^* = -1, \\ (Ax^0 - d)_i &= 0, \text{ if } -1 < y_i^* < 1. \end{aligned}$$

Let $0 < \gamma \leq \delta$, $x^0 \in X^0$, $x^\delta \in X^\delta$, and $\bar{x} = (1 - \frac{\gamma}{\delta})x^0 + \frac{\gamma}{\delta}x^\delta$. Then it follows from (15) and (16) that (14) holds. Since $A^T y^* = 0$, $-1 \leq y^* \leq 1$, (13) holds for $\bar{y} = y^*$. Thus, by Lemma 2.7, $\bar{x} \in X^\gamma$ and y^* is a solution of (10) for $0 < \gamma \leq \delta$. $\square$

The above lemma should be compared with Lemma 3 and Theorem 4 of [20] wherein results are formulated in terms of a “sign vector” associated with solutions to (12). The sign vector s^γ is related to y^γ as follows: $s_i^\gamma = y_i^\gamma$ if $|y_i^\gamma| = 1$ and $s_i^\gamma = 0$ if $|y_i^\gamma| < 1$. Madsen, Nielsen, and Pinar [20] proved that $s^\gamma \equiv s^\delta$ for $0 < \gamma < \delta$ if $\delta > 0$ is small enough.

3. Linear ℓ_1 estimator and Huber M-estimator. By Hoffman's error bound for approximate solutions of a system of linear equalities and inequalities, we obtain the Lipschitz stability of X^γ with respect to γ . By using the dual characterizations given in Corollaries 2.2 and 2.8, as well as Lemma 2.3, we prove that the solution set X^γ of (2) is a linear mapping of the parameter γ when $\gamma > 0$ is small enough. Moreover, we prove that the uniqueness of the Huber M-estimator for small positive tuning parameter γ implies the uniqueness of the linear ℓ_1 estimator. The converse is also true under the nondegeneracy assumption on the least norm solution of (3). But the converse is not true in general. We give an example of the linear ℓ_1 problem which has a unique solution, but the corresponding Huber M-estimator problem (2) does have infinitely many solutions for $0 < \gamma \leq \delta$ with some positive constant δ . Our example is based on a new explicit characterization of the least norm solution of (3).

First, we prove the Lipschitz stability of Huber M-estimators with respect to the perturbations of γ . Let X^γ be the solution set of (2) and X^0 be the solution set of (1). In order to prove the Lipschitz continuity of X^γ with respect to γ , we need the following lemma on Lipschitz continuity of solutions of linear equalities and inequalities [16, Theorem 2.4], which was based on Hoffman's error bound for approximate solutions of a system of linear equalities and inequalities (cf. [10, 23]). For two subsets X and Y of $\mathbf{R}^n$, the Hausdorff distance between X and Y is defined as follows:

$$H(X, Y) := \max \left\{ \max_{x \in X} \min_{y \in Y} \|x - y\|_2, \max_{y \in Y} \min_{x \in X} \|y - x\|_2 \right\}.$$

LEMMA 3.1. *Let $F(b, d) := \{x \in \mathbf{R}^n : Ax \leq b, Cx = d\}$. Then there is a positive constant η such that*

$$H(F(b, d), F(b', d')) \leq \eta(\|b - b'\| + \|d - d'\|)$$

whenever $F(b, d)$ and $F(b', d')$ are nonempty.

THEOREM 3.2. *Let $\delta > 0$ be such that the solution of (10) for $\gamma = \delta$ is the least norm solution y^* of (3). Then there exists a positive constant λ such that*

$$(17) \quad H(X^{\bar{\gamma}}, X^\gamma) \leq \lambda \cdot |\gamma - \bar{\gamma}| \quad \text{for } \gamma, \bar{\gamma} \in [0, \delta].$$

Proof. By Lemma 2.9, y^* is also the solution of (10) for $0 < \gamma < \delta$. Then it follows from Corollaries 2.2 and 2.8 that $\bar{x} \in X^\gamma$ if and only if

$$(18) \quad \begin{aligned} (A\bar{x} - d)_i &\geq \gamma, \text{ if } y_i^* = 1, \\ (A\bar{x} - d)_i &\leq -\gamma, \text{ if } y_i^* = -1, \\ (A\bar{x} - d)_i &= \gamma y_i^*, \text{ if } -1 < y_i^* < 1. \end{aligned}$$

By Lemma 3.1, there exists a positive constant η (depending only on A and y^*) such that

$$H(X^{\bar{\gamma}}, X^\gamma) \leq \eta \left[\left(\sum_{|y_i^*| < 1} (y_i^* \bar{\gamma} - y_i^* \gamma)^2 \right)^{\frac{1}{2}} + \left(\sum_{|y_i^*| = 1} (\bar{\gamma} - \gamma)^2 \right)^{\frac{1}{2}} \right] \leq \lambda |\bar{\gamma} - \gamma|,$$

where $\lambda = 2n\eta$. This completes the proof of Theorem 3.2. $\square$

Note that the above theorem shows that, for small γ , a Huber M-estimator is almost the same as a linear ℓ_1 estimator in the sense that the difference is a constant multiple of γ .

Various relations between X^γ and X^0 were given by Madsen, Nielsen, and Pinar [20]. However, these relations involve a sign pattern of certain vectors. Here we give a pure algebraic relation between X^γ and X^0 .

THEOREM 3.3. *Suppose that the solution of (10) for some $\gamma = \delta$ is the least norm solution y^* of (3). Then*

$$(19) \quad X^\tau \supset \frac{\tau - \alpha}{\beta - \alpha} X^\beta + \frac{\beta - \tau}{\beta - \alpha} X^\alpha \quad \text{for } 0 \leq \alpha < \tau < \beta \leq \delta,$$

where $\mu X + \nu Y := \{\mu x + \nu y : x \in X, y \in Y\}$.

Proof. By Lemma 2.9, y^* is the solution of (10) for $0 < \gamma \leq \delta$. Note that (14) reduces to (9) when $\gamma = 0$. Therefore, by Corollaries 2.2 and 2.8, $\bar{x} \in X^\gamma$ if and only if (14) holds, where $0 \leq \gamma \leq \delta$.

Let $0 \leq \alpha < \tau < \beta \leq \delta$, $x^\alpha \in X^\alpha$, $x^\beta \in X^\beta$, and $x^\tau := \frac{\tau - \alpha}{\beta - \alpha} x^\beta + \frac{\beta - \tau}{\beta - \alpha} x^\alpha$. Then (14) holds for $\gamma = \alpha$, $\bar{x} = x^\alpha$. Since $\frac{\tau - \alpha}{\beta - \alpha} > 0$, we get

$$(20) \quad \begin{aligned} \frac{\tau - \alpha}{\beta - \alpha} (Ax^\alpha - d)_i &\geq \frac{\tau - \alpha}{\beta - \alpha} \alpha, \text{ if } y_i^* = 1, \\ \frac{\tau - \alpha}{\beta - \alpha} (Ax^\alpha - d)_i &\leq -\frac{\tau - \alpha}{\beta - \alpha} \alpha, \text{ if } y_i^* = -1, \\ \frac{\tau - \alpha}{\beta - \alpha} (Ax^\alpha - d)_i &= \frac{\tau - \alpha}{\beta - \alpha} \alpha y_i^*, \text{ if } -1 < y_i^* < 1. \end{aligned}$$

Similarly, we have

$$(21) \quad \begin{aligned} \frac{\beta - \tau}{\beta - \alpha} (Ax^\beta - d)_i &\geq \frac{\beta - \tau}{\beta - \alpha} \beta, \text{ if } y_i^* = 1, \\ \frac{\beta - \tau}{\beta - \alpha} (Ax^\beta - d)_i &\leq -\frac{\beta - \tau}{\beta - \alpha} \beta, \text{ if } y_i^* = -1, \\ \frac{\beta - \tau}{\beta - \alpha} (Ax^\beta - d)_i &= \frac{\beta - \tau}{\beta - \alpha} \beta y_i^*, \text{ if } -1 < y_i^* < 1. \end{aligned}$$

Adding the corresponding equalities and inequalities in (20) and (21), respectively, we obtain (14) with $\gamma = \tau$ and $\bar{x} = x^\tau$. Thus, $x^\tau \in X^\tau$. Since x^α and x^β are arbitrary vectors in X^α and X^β , respectively, (19) holds. $\square$

The above theorem shows that X^γ is a convex set-valued mapping of γ for $0 \leq \gamma \leq \delta$. One important technical aspect of the above theorem is that δ must be small enough so that the solution of (10) for $\gamma = \delta$ is also a solution of (3). However, X^γ is actually a linear set-valued mapping of γ for $\gamma > 0$ small enough. The drawback of the next theorem is that there is no criterion to identify how small the γ should be.

THEOREM 3.4. *If the columns of A are linearly independent (i.e., the rank of A is n), then there exists a positive constant ϵ such that*

$$(22) \quad X^\tau = \frac{\tau - \alpha}{\beta - \alpha} X^\beta + \frac{\beta - \tau}{\beta - \alpha} X^\alpha \quad \text{for } 0 \leq \alpha < \tau < \beta \leq \epsilon.$$

Proof. Let $W_\gamma(x)$ be the diagonal matrix whose i th diagonal entry is 1 if $|(Ax - d)_i| \leq \gamma$ and is 0 otherwise. Then we can characterize an extreme point x of X^γ by the rank of $W_\gamma(x)A$: x is an extreme point of X^γ if and only if $W_\gamma(x^\gamma)A$ has rank n .

Recall that x is an extreme point of X^γ if and only if $x = \lambda y + (1 - \lambda)z$ implies $x = y = z$ whenever $y, z \in X^\gamma$ and $0 < \lambda < 1$ [27, page 162].

Suppose that $x \in X^\gamma$ and $W_\gamma(x)A$ has rank less than n . Then there exists a vector $u \neq 0$ such that $W_\gamma(x)Au = 0$. Let σ be a small positive constant such that

$(A(x \pm \sigma u) - d)_i > \gamma$ if $(Ax - d)_i > \gamma$ and $(A(x \pm \sigma u) - d)_i < -\gamma$ if $(Ax - d)_i < -\gamma$. Then

$$A^T(A(x \pm \sigma u) - d)_{-\gamma}^\gamma = A^T(Ax - d)_{-\gamma}^\gamma = 0,$$

and so $(x \pm \sigma u) \in X^\gamma$. Since $x = \frac{1}{2}(x + \sigma u) + \frac{1}{2}(x - \sigma u)$ and $u \neq 0$, x is not an extreme point of X^γ .

On the other hand, suppose that $x \in X^\gamma$ and $W_\gamma(x)A$ has rank n . If $y, z \in X^\gamma$ such that $x = \lambda y + (1 - \lambda)z$ with some $0 < \lambda < 1$, then it follows from (14) that $(Ax - d)_i = y_i^* \gamma$ if and only if $(Ay - d)_i = (Az - d)_i = y_i^* \gamma$. Thus,

$$W_\gamma(x)Ax = W_\gamma(x)Ay = W_\gamma(x)Az.$$

Since $W_\gamma(x)A$ has rank n , we have $x = y = z$. By the definition, x is an extreme point of X^γ .

Now let $\delta > 0$ be such that the solution of (10) for $\gamma = \delta$ is the least norm solution y^* of (3).

Suppose $\alpha < \beta \leq \delta$ and $\tau = \lambda\alpha + (1 - \lambda)\beta$ with $0 < \lambda < 1$. Let x^α and x^β be extreme points of X^α and X^β , respectively. By Theorem 3.3, $x^\tau \in X^\tau$ with $x^\tau := \lambda x^\alpha + (1 - \lambda)x^\beta$. If $W_\alpha(x^\alpha) = W_\beta(x^\beta)$, then it is easy to verify that $W_\tau(x^\tau) = W_\alpha(x^\alpha) = W_\beta(x^\beta)$. Therefore, x^τ is an extreme point of X^τ . In other words, if $W \in \mathcal{W}_\beta \cap \mathcal{W}_\alpha$ with $\alpha < \beta$, then $W \in \mathcal{W}_\tau$ for $\alpha < \tau < \beta$, where

$$\mathcal{W}_\gamma := \{W_\gamma(x) : x \text{ is an extreme point of } X^\gamma\}.$$

Let $W \in \bigcup_{\gamma > 0} \mathcal{W}_\gamma$. Then, by the above argument, the set $\{\gamma : W \in \mathcal{W}_\gamma\}$ is a segment of line. Therefore, there is some small positive constant ϵ such that either

$$(23) \quad W \in \mathcal{W}_\gamma \quad \text{for } 0 < \gamma \leq \epsilon$$

or

$$(24) \quad W \notin \mathcal{W}_\gamma \quad \text{for } 0 < \gamma \leq \epsilon.$$

Since $\bigcup_{\gamma > 0} \mathcal{W}_\gamma$ is a finite set, we can choose $0 < \epsilon \leq \delta$ such that either (23) or (24) holds for every $W \in \bigcup_{\gamma > 0} \mathcal{W}_\gamma$.

Define $\mathcal{W}^* := \{W \in \bigcup_{\gamma > 0} \mathcal{W}_\gamma : (23) \text{ holds}\}$. Let $0 < \alpha < \beta \leq \epsilon$ and $\alpha < \tau < \beta$. Let x^τ be an extreme point of X^τ . Then $W := W_\tau(x^\tau) \in \mathcal{W}^*$. By the definition of $\mathcal{W}^*$, there exist $x^\alpha \in X^\alpha$ and $x^\beta \in X^\beta$ such that $W = W_\alpha(x^\alpha) = W_\beta(x^\beta)$. Note that we have proved in the preceding paragraph that $\bar{x}^\tau \in X^\tau$ and $W_\alpha(\bar{x}^\tau) = W$, where

$$\bar{x}^\tau := \frac{\beta - \tau}{\beta - \alpha} x^\alpha + \frac{\tau - \alpha}{\beta - \alpha} x^\beta.$$

Thus, $WA\bar{x}^\tau = WAx^\tau$. Since WA has rank n , $\bar{x}^\tau = x^\tau$. That is,

$$(25) \quad x^\tau = \frac{\beta - \tau}{\beta - \alpha} x^\alpha + \frac{\tau - \alpha}{\beta - \alpha} x^\beta.$$

Therefore, extreme points of X^τ are in the set

$$\bar{X} := \frac{\beta - \tau}{\beta - \alpha} X^\alpha + \frac{\tau - \alpha}{\beta - \alpha} X^\beta.$$

Since the rank of A is n , the column vectors of A are linearly independent. Thus, $\|Ax\| \rightarrow \infty$ as $\|x\| \rightarrow \infty$. As a consequence, $\sum_{i=1}^m \rho[(Ax - d)_i] \rightarrow \infty$ as $\|x\| \rightarrow \infty$, which implies that X^γ is bounded. Since X^γ is also the set of dual solutions of the strictly convex quadratic program (10), X^γ is a convex polyhedral subset of $\mathbf{R}^n$. Since any point in a compact polyhedron X^τ is a convex combination of extreme points of X^τ [27, Corollary 18.5.1], any point in X^τ is a convex combination of vectors in $\bar{X}$. Since $\bar{X}$ is convex, we get

$$(26) \quad X^\tau \subset \frac{\beta - \tau}{\beta - \alpha} X^\alpha + \frac{\tau - \alpha}{\beta - \alpha} X^\beta.$$

The linearity property (22) with $\alpha > 0$ follows from (26) and (19).

Let $\alpha \rightarrow 0^+$ in (22). By Theorem 3.2, $X^\alpha \rightarrow X^0$. Therefore, (22) also holds when $\alpha = 0$. $\square$

Remark. The assumption that A has rank n is not necessary in Theorem 3.4. In fact, if the rank of A is $\hat{n}$ (less than n), without loss of generality we may assume that the first $\hat{n}$ columns of A are linearly independent. Let $\hat{A}$ be the matrix consisting of the first $\hat{n}$ columns of A . Then the columns $\hat{A}$ generate the same subspace as the columns of A . Thus, the Huber M-estimator problem 12 is reduced to the following one:

$$(27) \quad \min_{\hat{x} \in \mathbf{R}^{\hat{n}}} \sum_{i=1}^m \rho[(\hat{A}\hat{x} - d)_i].$$

Let $\hat{X}^\gamma$ be the solution set of (27). By Theorem 3.4, $\hat{X}^\gamma$ is affine when γ is sufficiently small. However, it is not difficult to verify that

$$X^\gamma = \{x \in \mathbf{R}^n : (x_1, \dots, x_{\hat{n}})^T \in \hat{X}^\gamma, x_{\hat{n}+1} = \dots = x_n = 0\} + \mathcal{N},$$

where $\mathcal{N} := \{x \in \mathbf{R}^n : Ax = 0\}$ is the null space of A . Since $\mathcal{N}$ is a subspace, X^γ is also affine in the γ range where $\hat{X}^\gamma$ is affine.

Note that if columns of A are not linearly independent, then X^γ is an unbounded set and X^γ is not the convex hull of its extremal points. In this case, the proof of Theorem 3.4 fails.

Let $\alpha = 0$, $\beta = \delta$, and $0 < \tau < \delta$ in Theorem 3.3. By (19), we have $X^\tau \supset \frac{\tau}{\delta} X^\delta + (1 - \frac{\tau}{\delta}) X^0$. If $X^\tau = \{x^\tau\}$ has only one element, then $X^0 = \{x^0\}$ and $X^\delta = \{x^\delta\}$ are both singletons. Moreover, $x^\tau = \frac{\tau}{\delta} x^\delta + (1 - \frac{\tau}{\delta}) x^0$. That is, x^τ is a linear function of τ for $0 \leq \tau \leq \delta$. Thus, we have the following consequence of Theorem 3.3.

THEOREM 3.5. *Suppose that the solution of (10) for some $\gamma = \delta$ is the least norm solution of (3). If the Huber M-estimator problem (2) has a unique solution x^γ for $\gamma > 0$ small enough, then the linear ℓ_1 estimation problem (1) has a unique solution x^0 . Moreover, $x^\gamma = \frac{\gamma}{\delta} x^\delta + (1 - \frac{\gamma}{\delta}) x^0$ is a linear function of γ for $0 \leq \gamma \leq \delta$.*

It is interesting to see that $x^0 = \frac{\delta}{\delta - \gamma} x^\gamma - \frac{\gamma}{\delta - \gamma} x^\delta$ for $0 < \gamma < \delta$. That is, we can find the unique linear ℓ_1 estimator by two Huber M-estimators of different small tuning parameters.

The converse of the above theorem about the uniqueness is also true if the least norm solution of (3) is nondegenerate. A solution y^* of (3) is said to be nondegenerate if there exists a solution $\bar{x}$ of (1) such that

$$(28) \quad \begin{aligned} (A\bar{x} - d)_i &> 0, \text{ if } y_i^* = 1, \\ (A\bar{x} - d)_i &< 0, \text{ if } y_i^* = -1, \\ (A\bar{x} - d)_i &= 0, \text{ if } -1 < y_i^* < 1. \end{aligned}$$

THEOREM 3.6. *Suppose that the least norm solution of (3) is nondegenerate and the linear ℓ_1 estimation problem (1) has a unique solution. Then there exists $\delta > 0$ such that the Huber M-estimator problem (2) has a unique solution x^γ for $0 < \gamma < \delta$. Moreover, the objective function $\Phi(x) = \sum_{i=1}^m \rho[(Ax - d)_i]$ in (2) is actually a strictly convex quadratic function in a neighborhood of x^γ for $0 < \gamma \leq \delta$ and $\Phi(x)$ is twice differentiable at x^γ (i.e., the Hessian $\Phi''(x^\gamma)$ exists).*

Proof. Let $\delta_0 > 0$ be such that the solution of y^γ of (10) is the least norm solution y^* of (3) for $0 < \gamma \leq \delta_0$. Let $\bar{x}$ be the unique solution of (1). Then (28) holds. Let W be the diagonal matrix whose i th diagonal entry is 1 if $-1 < y_i < 1$ and 0 otherwise. Then we claim that the homogeneous linear system $WAz = 0$ has a unique solution $z^* = 0$.

In fact, assume $WA\bar{z} = 0$. By (28), there exists $\epsilon > 0$ (small enough) such that

$$\begin{aligned} (A(\bar{x} + \epsilon\bar{z}) - d)_i &> 0, \text{ if } y_i^* = 1, \\ (A(\bar{x} + \epsilon\bar{z}) - d)_i &< 0, \text{ if } y_i^* = -1, \\ (A(\bar{x} + \epsilon\bar{z}) - d)_i &= 0, \text{ if } -1 < y_i^* < 1. \end{aligned}$$

By Corollary 2.2, $(\bar{x} + \epsilon\bar{z})$ is a solution of (1). By the uniqueness of $\bar{x}$, $\bar{x} = \bar{x} + \epsilon\bar{z}$, which implies $\bar{z} = 0$. This proves our claim.

Now, for $0 < \gamma \leq \delta_0$, let x^γ and $\bar{x}^\gamma$ be solutions of (2). Since $y^\gamma \equiv y^*$, by Lemma 2.6, $y^* = \frac{1}{\gamma}(Ax^\gamma - d)_{-\gamma}^\gamma = \frac{1}{\gamma}(A\bar{x}^\gamma - d)_{-\gamma}^\gamma$. Thus, for $-1 < y_i^* < 1$,

$$((Ax^\gamma - d)_{-\gamma}^\gamma)_i = ((A\bar{x}^\gamma - d)_{-\gamma}^\gamma)_i = \gamma y_i^*,$$

which implies

$$W(Ax^\gamma - d)_{-\gamma}^\gamma = W(Ax^\gamma - d) \text{ and } W(A\bar{x}^\gamma - d)_{-\gamma}^\gamma = W(A\bar{x}^\gamma - d).$$

Therefore, $W(Ax^\gamma - d) = W(\gamma y^*) = W(A\bar{x}^\gamma - d)$; i.e., $WA(x^\gamma - \bar{x}^\gamma) = 0$. Hence, $x^\gamma = \bar{x}^\gamma$ by our claim and (2) has a unique solution for $0 < \gamma \leq \delta_0$.

Finally, let x^γ be the unique solution of (2) for $0 < \gamma \leq \delta_0$. By Theorem 3.2, $\lim_{\gamma \rightarrow 0^+} (Ax^\gamma - d - \gamma y^*) = A\bar{x} - d$. By (28), there exists $0 < \delta \leq \delta_0$ such that, for $0 < \gamma \leq \delta$,

$$\begin{aligned} (Ax^\gamma - d - \gamma)_i &> 0, \text{ if } y_i^* = 1, \\ (Ax^\gamma - d + \gamma)_i &< 0, \text{ if } y_i^* = -1. \end{aligned}$$

Since $y^\gamma = y^*$, by Corollary 2.8, we also have $-\gamma < \gamma y_i^* = (Ax^\gamma - d)_i < \gamma$ whenever $-1 < y_i^* < 1$. As a consequence, for each $0 < \gamma \leq \delta$, there exists a neighborhood $U(\gamma)$ of x^γ in $\mathbf{R}^n$ such that

$$\begin{aligned} (Ax - d)_i &> \gamma, \text{ if } y_i^* = 1, \\ (29) \quad (Ax - d)_i &< -\gamma, \text{ if } y_i^* = -1, \quad \text{for } x \in U(\gamma). \\ -\gamma &< (Ax - d)_i < \gamma, \text{ if } -1 < y_i^* < 1, \end{aligned}$$

Therefore, for $x \in U(\gamma)$,

$$\Phi'(x) = A^T(Ax - d)_{-\gamma}^\gamma = A^T W(Ax - d) + A^T(I - W)(\gamma y^*),$$

which implies that $\Phi(x)$ is a quadratic function in $U(\gamma)$ and $\Phi''(x) = A^T W A$ exists. Since $WAz = 0$ has a unique solution $z^* = 0$, $A^T W A$ is a positive definite matrix. Thus, $\Phi(x)$ is actually a strictly convex quadratic function in $U(\gamma)$. $\square$

Remark. Let y^γ be the solution of (10) and $W(\gamma)$ be the diagonal matrix whose i th diagonal entry is 1 if $-1 < y_i^\gamma < 1$ and 0 otherwise. Since $W(\gamma)Az = 0$ has a unique solution $z^* = 0$ whenever $A^T W(\gamma)A$ is nonsingular, the above proof implies that (2) has a unique solution if $A^T W(\gamma)A$ is nonsingular. This sufficient condition for uniqueness of the Huber M-estimator was observed by many authors [5, 18, 19, 20].

However, the above theorem is not true without the nondegeneracy assumption. To construct a counterexample, we need the following explicit characterization of the least norm solution of (3).

THEOREM 3.7. *Let $\varphi \in \mathbf{R}^m$ be such that $\varphi^T A = 0$, $\max_{1 \leq i \leq m} |\varphi_i| = 1$, and there exists a positive constant δ such that, for $0 < \gamma \leq \delta$,*

$$X^\gamma(\varphi) := \left\{ x \in \mathbf{R}^n : \begin{array}{l} \varphi_i(Ax - d)_i \geq \gamma \text{ if } |\varphi_i| = 1, \\ (Ax - d)_i = \varphi_i \gamma \text{ if } |\varphi_i| < 1 \end{array} \right\} \neq \emptyset.$$

Then φ is the least norm solution of (3) and $X^\gamma(\varphi) \equiv X^\gamma$ for $0 < \gamma \leq \delta$.

Proof. Let $x^\gamma \in X^\gamma(\varphi)$. Then $\varphi = \frac{1}{\gamma}(Ax^\gamma - d)_{-\gamma}^\gamma$ and

$$A^T(Ax^\gamma - d)_{-\gamma}^\gamma = \gamma A^T \varphi = \gamma(\varphi^T A)^T = 0.$$

Therefore, by the equivalent form (12) of (2), x^γ is a solution of (2). Then, from Lemma 2.6, we know that $\varphi = \frac{1}{\gamma}(Ax^\gamma - d)_{-\gamma}^\gamma$ is a solution of (10) for $0 < \gamma \leq \delta$. By Lemma 2.3, φ is the least norm solution of (3). Finally, it follows from Corollary 2.8 that $X^\gamma(\varphi) \equiv X^\gamma$. $\square$

Example 3.8. Let $\alpha \geq 0$,

$$A = \begin{pmatrix} 1 & 4(\alpha + 1) \\ 1 & -4(\alpha + 1) \\ 0 & 2 \\ 0 & 9 + 8\alpha \end{pmatrix}, \quad \text{and} \quad d = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

Then (1) has a unique solution $x^* = 0$. However, the corresponding Huber M-estimator problem has the following solution set:

$$X^\gamma = \left\{ x \in \mathbf{R}^2 : -\alpha\gamma \leq x_1 \leq \alpha\gamma, x_2 = \frac{\gamma}{4} \right\} \quad \text{for } 0 < \gamma \leq \frac{4}{13 + 8\alpha}$$

which contains infinitely many solutions of (2) when $\alpha > 0$.

Proof. Let

$$\varphi = \begin{pmatrix} 1 \\ -1 \\ \frac{1}{2} \\ -1 \end{pmatrix}.$$

Then $\|\varphi\|_\infty = 1$ and $\varphi^T A = 0$. Consider the set $X^\gamma(\varphi)$ defined by the following system of inequalities and equalities:

$$\begin{aligned} (30) \quad & x_1 + 4(\alpha + 1)x_2 \geq \gamma, \\ & x_1 - 4(\alpha + 1)x_2 \leq -\gamma, \\ & 2x_2 = \frac{1}{2}\gamma, \\ & (9 + 8\alpha)x_2 - 1 \leq -\gamma. \end{aligned}$$

From the third equation in (30) we get $x_2 = \frac{\gamma}{4}$. Since $0 < \gamma \leq \frac{4}{13+8\alpha}$, the last inequality in (30) is also true for any x_1 . Substituting $x_2 = \frac{\gamma}{4}$ into the first two inequalities in (30) and solving for x_1 , we obtain $-\alpha\gamma \leq x_1 \leq \alpha\gamma$. Therefore, $X^\gamma(\varphi) \neq \emptyset$ for $0 < \gamma \leq \frac{4}{13+8\alpha}$. By Theorem 3.7, φ is the least norm solution of (3) and $X^\gamma = X^\gamma(\varphi)$. Since $x^\gamma \rightarrow 0$ as $\gamma \rightarrow 0$ for any $x^\gamma \in X^\gamma$, by Theorem 3.2, $x^* = 0$ is the only solution of (1). $\square$

Note that in the above example, the least norm solution φ of (3) is degenerate, even though (3) has nondegenerate solutions. In the case that $\alpha = 0$, the Huber M-estimator problem (2) also has a unique solution x^γ with $x_1^\gamma = 0$ and $x_2^\gamma = \frac{\gamma}{4}$ for small γ , despite the degeneracy of the least norm solution φ . Therefore, the nondegeneracy assumption in Theorem 3.6 is not necessary for the uniqueness of the Huber M-estimator.

4. Computation of a linear ℓ_1 estimator. In [19], Madsen and Nielsen proposed to compute a sign pattern vector s for the linear ℓ_1 estimation problem (1) by solving (2) for very small tuning parameter γ . Then, by solving a system of linear inequalities and equalities, a linear ℓ_1 estimator can be obtained. In this section we propose a recursive version of Madsen and Nielsen's algorithm for finding a linear ℓ_1 estimator. By the dual relationships given in section 2, we know exactly how small γ should be in computation. Moreover, unlike Madsen and Nielsen's approach, a linear ℓ_1 estimator is explicitly constructed by using Huber M-estimators.

First let us explain Madsen and Nielsen's method for computing a linear ℓ_1 estimator by solving Huber M-estimator problems. Suppose that we have a solution x^γ of (12) (or (2)) for some very small γ . By Lemma 2.6, $y^\gamma = \frac{1}{\gamma}(Ax^\gamma - d)_{-\gamma}^\gamma$ is a solution of (10). Then, by Lemma 2.3, y^γ is the least norm solution of (3). By Corollary 2.2, one can get a linear ℓ_1 estimator by solving the feasibility problem (9), a system of linear inequalities and equalities with $y^* \equiv y^\gamma$.

In general, a system of linear inequalities and equalities is not trivial to solve. However, under the assumption of nonsingularity of a certain matrix, (9) is reduced to a system of linear equations. More specifically, let $W(\gamma)$ be the $m \times m$ diagonal matrix whose i th diagonal entry is 1 if $|y_i^\gamma| < 1$ and 0 if $|y_i^\gamma| = 1$. By Lemma 2.3, there exists a positive number δ such that $y^\gamma \equiv y^0$, the least norm solution of (3), for $0 < \gamma < \delta$. As a result, $W(\gamma) \equiv W(0)$ is a constant matrix for $0 < \gamma < \delta$. If $A^T W(\gamma) A$ is nonsingular, then the unique solution of (9) can be found by solving a system of linear equations as shown in the following theorem.

LEMMA 4.1. *Let $\delta > 0$ be such that the solution y^δ of (10) (with $\gamma = \delta$) is the least norm solution of (3), and let $W(\delta)$ be the $m \times m$ diagonal matrix whose i th diagonal entry is 1 if $|y_i^\delta| < 1$ and 0 if $|y_i^\delta| = 1$. If $A^T W(\delta) A$ is nonsingular, then the Huber M-estimator problem (2) has a unique solution x^γ for $0 < \gamma < \delta$ and*

$$(31) \quad x^\gamma = x^\delta + (\gamma - \delta)z^0,$$

where z^0 is the unique solution of the following linear system:

$$(32) \quad W(\delta)Az = W(\delta)y^\delta.$$

Moreover, the linear ℓ_1 estimation problem (1) has a unique solution $x^0 = x^\delta - \delta z^0$, which is also the unique solution of the following linear system:

$$(33) \quad W(\delta)Ax = W(\delta)d.$$

Proof. By Lemma 2.9, for $0 < \gamma \leq \delta$, y^γ is also the least norm solution of (3) and $W(\gamma) = W(\delta)$. Thus, $A^T W(\gamma) A$ is nonsingular and, by the remark after Theorem 3.6, (2) has a unique solution x^γ . By Theorem 3.5, (1) has a unique solution x^0 . By Corollary 2.8,

$$(34) \quad W(\delta)(Ax^\gamma - d) = W(\gamma)(Ax^\gamma - d) = \gamma W(\gamma)y^\gamma = \gamma W(\delta)y^\delta.$$

By Theorem 3.2, $\lim_{\gamma \rightarrow 0^+} x^\gamma = x^0$. Thus, it follows from (34) that

$$(35) \quad W(\delta)Ax^0 = W(\delta)d.$$

Since $A^T W(\delta)A$ is nonsingular, $W(\delta)Az = 0$ has a unique solution $z^* = 0$. Thus, x^0 is the unique solution of (33). It follows from (34) and (35) that

$$W(\delta)A \left(\frac{x^\gamma - x^0}{\gamma} \right) = W(\delta)y^\delta.$$

Since (32) has at most one solution, $z^0 := (\frac{x^\gamma - x^0}{\gamma})$ is the unique solution of (32) and $x^0 = x^\gamma - \gamma z^0$ for $0 < \gamma \leq \delta$. Finally, (31) follows from $x^\delta - \delta z^0 = x^0 = x^\gamma - \gamma z^0$ for $0 < \gamma \leq \delta$. $\square$

In general, from (9) we know that all solutions of (1) are in the set

$$X(y^\gamma) := \{x \in \mathbf{R}^n : (Ax - d)_i = 0 \text{ if } |y_i^\gamma| < 1\}.$$

Therefore, we consider the following convex differentiable minimization problem:

$$(36) \quad \min_{x \in X(y^\gamma)} \sum_{i=1}^m \rho[(Ax - d)_i].$$

Let $x^\gamma \in X(y^\gamma)$. Then $[X(y^\gamma) - x^\gamma]$ is a linear subspace of $\mathbf{R}^n$. Let $\bar{n} = \dim[X(y^\gamma) - x^\gamma] < n$. Then there is an $n \times \bar{n}$ matrix T such that

$$[X(y^\gamma) - x^\gamma] = \{T\bar{x} : \bar{x} \in \mathbf{R}^{\bar{n}}\}.$$

Thus, (36) can be rewritten as

$$\min_{\bar{x} \in \mathbf{R}^{\bar{n}}} \sum_{i=1}^m \rho[(A(x^\gamma + T\bar{x}) - d)_i]$$

or

$$(37) \quad \min_{\bar{x} \in \mathbf{R}^{\bar{n}}} \sum_{i=1}^m \rho[(\bar{A}\bar{x} - \bar{d})_i],$$

where $\bar{A} = AT$ and $\bar{d} = d - Ax^\gamma$. Note that (37) is the Huber M-estimator problem (cf. (12)) corresponding to a “simpler” linear ℓ_1 estimation problem:

$$(38) \quad \min_{\bar{x} \in \mathbf{R}^{\bar{n}}} \sum_{i=1}^m |(\bar{A}\bar{x} - \bar{d})_i|.$$

One important relation between these two ℓ_1 estimation problems is that if $\bar{x}^*$ is a solution of the new ℓ_1 estimation problem (38), then $x^* = x^\gamma + T\bar{x}^*$ is a solution of

the original ℓ_1 estimation problem (1). Therefore, the problem is reduced to finding a solution of a “simpler” problem. After repeating this process a few times, we prove that either the condition in Lemma 4.1 will be satisfied for the final Huber M-estimator problem or all the common zero indices of the residual vectors will be identified. Then we can get a solution of the final linear ℓ_1 estimation problem either by solving a system of linear equations or by solving another Huber M-estimator problem. By doing so, we can explicitly construct a linear ℓ_1 estimator of (1) by using the Huber M-estimators and transformation matrices obtained in the process.

The above recursive method of computing a linear ℓ_1 estimator is formulated in the following algorithm.

ALGORITHM 4.2. Let $k = 0$, $T^0 = I$ (the $n \times n$ identity matrix), $d^0 = d$, $x^0 = 0$, and $A^0 = A$. Compute a solution of (1) as follows:

Step 1. find $\gamma > 0$ and a solution u^k of

$$(39) \quad (A^k)^T (A^k u - d^k)_{-\gamma}^\gamma = 0$$

such that $v^k := \frac{1}{\gamma}(A^k u^k - d^k)_{-\gamma}^\gamma$ is a solution of

$$(40) \quad \min_v (d^k)^T v \quad \text{subject to} \quad (A^k)^T v = 0, -1 \leq v \leq 1;$$

Step 2. let W^k be the diagonal matrix whose i th diagonal entry is 1 if $|v_i^k| < 1$ and 0 otherwise;

Step 3. if $k > 0$ and $W^k = W^{k-1}$, let $x^{k+1} := x^k + T^k u^k$ and go to **Step 9**;

Step 4. find a solution z^k of the linear system

$$(41) \quad W^k A^k z = W^k d^k;$$

Step 5. let $x^{k+1} := x^k + T^k z^k$;

Step 6. if $(A^k)^T W^k A^k$ is nonsingular, then go to **Step 9**;

Step 7. let the columns of B^k be a basis of the null space of $W^k A^k$;

Step 8. set $A^{k+1} := A^k B^k$, $T^{k+1} = T^k B^k$, $d^{k+1} := d^k - A^k z^k$, $k := k + 1$, and go to **Step 1**;

Step 9. output x^{k+1} as a solution of (1).

Proof. First we show that $W^k = W^{k-1}$ for some integer k . Since $W^{k-1} A^k = 0$ and $W^{k-1} d^k = 0$, we have $W^{k-1} v^k = 0$. As a consequence, $W^{k-1} \leq W^k$ in the sense that if the i th diagonal element of W^{k-1} is 1 then the i th diagonal element of W^k is 1. Therefore, $W^k = W^{k-1}$ for some $0 < k \leq m + 1$. Thus, the algorithm terminates in finitely many iterations.

Now we claim that if q^k is a solution of the linear ℓ_1 estimation problem

$$(42) \quad \min_z \|A^k z - d^k\|_1,$$

then $x^{k+1} = x^k + T^k q^k$ is a solution of (1).

By Corollary 2.2, $W^k(A^k \bar{z} - d^k) = 0$ for any solution $\bar{z}$ of (42). Thus, (41) is consistent. Moreover, $\bar{z}$ is a solution of (42) if and only if $\bar{z} = z^k + u$, where $W^k A^k u = 0$. Let $d^{k+1} := d^k - A^k z^k$ and consider the problem

$$(43) \quad \min_u \|A^k u - d^{k+1}\|_1 \quad \text{subject to} \quad W^k A^k u = 0.$$

The minimization problem (43) can be reformulated as

$$(44) \quad \min_v \|A^k B^k v - d^{k+1}\|_1,$$

where the columns of B^k form a basis for the null space of $W^k A^k$. Note that q^k is a solution of (42) if and only if $q^k = z^k + B^k q^{k+1}$, where q^{k+1} is a solution of (44).

We prove the claim by induction. It is obvious that the claim holds if $k = 0$. We show that the claim holds for $(k + 1)$ under the assumption that the claim holds for k . Let q^{k+1} be a solution of (44). Then $q^k := z^k + B^k q^{k+1}$ is a solution of (42). Thus,

$$\begin{aligned} x^{k+1} + T^{k+1} q^{k+1} &= (x^k + T^k z^k) + T^k B^k q^{k+1} \\ &= x^k + T^k (z^k + B^k q^{k+1}) = x^k + T^k q^k, \end{aligned}$$

which is a solution of (1) by the inductive hypothesis. This proves our claim.

Finally, we show that the algorithm produces a solution of (1). Observe that, when $W^{k-1} = W^k$, $|v_i^k| < 1$ if and only if $A_i^k = 0$ and $d_i^k = 0$. As a consequence (cf. Corollary 2.8),

$$\begin{aligned} (A^k u^k - d^k)_i &\geq \gamma, & \text{if } v_i^k &= 1, \\ (A^k u^k - d^k)_i &\leq -\gamma, & \text{if } v_i^k &= -1, \\ (A^k u^k - d^k)_i &= 0, & \text{if } -1 < v_i^k < 1. \end{aligned}$$

By Corollary 2.2, u^k is a solution of the linear ℓ_1 estimation problem (42). By our claim, $x^{k+1} := x^k + T^k u^k$ is a solution of (1). If $(A^k)^T W^k A^k$ is nonsingular, by Lemma 4.1, z^k is the unique solution of (42). Again, by our claim, $x^{k+1} := x^k + T^k z^k$ is a solution of (1). Thus, the algorithm produces a solution of (1). $\square$

Remark. One can prove that $W^k = W^{k+1}$ if and only if the following condition holds: $(A^k \bar{z} - d^k)_i = 0$ for all solutions $\bar{z}$ of (42) if and only if the i th entry of W^{k-1} is 1. That is, $W^k = W^{k+1}$ if and only if W^k “identifies” the common zero indices of the residuals $(A^k \bar{z} - d^k)$ for all solutions $\bar{x}$ of (42). If the least norm solution of (3) is nondegenerate, then $W^1 = W^0$ and we can find a linear ℓ_1 estimator by solving two Huber M-estimator problems. It is interesting to see that if $W^k = W^{k-1}$, the Huber M-estimator u^k is also a solution of the corresponding linear ℓ_1 estimation problem (42). In the degenerate case, we say that i is a degenerate index if $(A\bar{x} - d)_i = 0$ for all solutions $\bar{x}$ of (1), but the i th component of the least norm solution of (3) is either 1 or -1 . Then the recursive process “tries” to identify which indices among the “degenerate” indices are the common zero indices of the residuals $(A\bar{x} - d)$ for all solutions $\bar{x}$ of (1). Once all such indices are identified, $W^k = W^{k-1}$ and the algorithm will produce a solution of (1) in the next iteration. However, if “enough” common zero indices are identified in the current iteration (i.e., $(A^k)^T W^k A^k$ is nonsingular), then we know from Lemma 4.1 that a linear ℓ_1 estimator can be constructed by solving (41). Therefore, **Step 6** gives us a possible early termination of the algorithm before we identify all common zero indices.

Let us apply Algorithm 4.2 to the degenerate linear ℓ_1 estimation problem in Example 3.8 and try to understand how degeneracy is resolved. Let $\alpha \geq 1$ and consider the following linear ℓ_1 estimation problem:

$$\min_x \|Ax - d\|_1 = \min_{x_1, x_2} \left\| \begin{pmatrix} 1 & 4(\alpha + 1) \\ 1 & -4(\alpha + 1) \\ 0 & 2 \\ 0 & 9 + 8\alpha \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \right\|_1.$$

Step 1 in Algorithm 4.2 is the most difficult step. One has to use another algorithm (such as a Newton method, etc. [6, 7, 24, 18, 29, 13, 14, 15]) to find a solution of (39).

Note that v^k is a feasible approximate solution of (40) and it is possible to use primal active set method to verify whether v^k is a solution of (40) or not. In general, we may have to try a few different γ s before v^k becomes a solution of (40). In the following example, we will use a small γ so that v^k is a solution of (40). In our calculation we actually use Theorem 3.7 to find x^γ and the least norm solution. Let $x^0 = 0$, $A^0 = A$, $d^0 = d$, and $T^0 = I$.

Steps 1–2. Let $\gamma = \frac{4}{13+8\alpha}$ and solve the corresponding Huber M-estimator problem:

$$\begin{pmatrix} 1 & 4(\alpha+1) \\ 1 & -4(\alpha+1) \\ 0 & 2 \\ 0 & 9+8\alpha \end{pmatrix}^T \left(\begin{pmatrix} 1 & 4(\alpha+1) \\ 1 & -4(\alpha+1) \\ 0 & 2 \\ 0 & 9+8\alpha \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \right)_{-\gamma}^\gamma = 0.$$

By Example 3.8, we can get a solution of the above system as follows:

$$u_1^\gamma = \beta\gamma, \quad u_2^\gamma = \frac{\gamma}{4},$$

where $-\alpha \leq \beta \leq \alpha$. Note that

$$v^0 = \begin{pmatrix} 1 \\ -1 \\ \frac{1}{2} \\ -1 \end{pmatrix} \quad \text{and} \quad W^0 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Step 3. Since $k = 0$, we move to **Step 4**.

Step 4. The system (41) becomes $2z_2 = 0$. If we use pseudoinverse to solve (41) (i.e., we get the least norm solution of (41)), then $z_1^0 = z_2^0 = 0$.

Step 5. $x^1 := x^0 + T^0 z^0 = 0$.

Step 6. $(A^0)^T W^0 A^0 = \begin{pmatrix} 0 & 0 \\ 0 & 4 \end{pmatrix}$ is singular.

Step 7. Since $W^0 A^0 = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 2 \\ 0 & 0 \end{pmatrix}$, we choose $B^0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ with its column as a basis of the null space of $W^0 A^0$.

Step 8. Compute $A^1 = A^0 B^0 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$, $T^1 = T^0 B^0 = B^0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $d^1 = d^0 - A^0 z^0 =$

$$d^0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \text{ and go back to Step 1.}$$

Steps 1–2. Consider the following new Huber M-estimator problem:

$$\begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}^T \left(\begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} u - \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \right)_{-\gamma}^\gamma = 0.$$

For any $0 < \gamma \leq 1$, $u^\gamma = 0$,

$$v^1 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \end{pmatrix} \quad \text{and} \quad W^1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Note that the associated linear ℓ_1 estimation problem is the following one:

$$\min_u \left\| \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} u - \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \right\|_1.$$

Step 3. Since $W^1 \neq W^0$, we move to **Step 4**.

Step 4. $W^1 A^1 z = W^1 d^1$ becomes $z = 0$ and it has a unique solution $z^1 = 0$.

Step 5. Let $x^2 = x^1 + T^1 z^1 = 0$.

Step 6. Since $(A^1)^T W^1 A^1 = (2)$ is a nonsingular 1×1 matrix, we skip to **Step 9** and output $x^2 = 0$ as a linear ℓ_1 estimator.

5. Conclusions. We show that many new relationships between the linear ℓ_1 estimators and the Huber M-estimators can be derived once one understands that the dual solution of the Huber M-estimator problem (2) with a small tuning parameter γ is the least norm solution of the dual problem of the linear ℓ_1 estimation problem (1). The dual connection between (2) and (1) allows us to establish a set-valued version of the linearity property of the Huber M-estimator with respect to the tuning parameter γ , as well as the Lipschitz stability of the Huber M-estimators with respect to perturbations of γ . It also allows us to identify how small the tuning parameter should be in the computation of the Huber M-estimators for finding a linear ℓ_1 estimator. By exploiting the dual connection, we have an almost complete understanding of how the uniqueness of the Huber M-estimator with a small tuning parameter is related to the uniqueness of the linear ℓ_1 estimator.

Our study also provides a new interpretation of the role of the Huber M-estimator in Madsen and Nielsen's method for computing a linear ℓ_1 estimator. In the case that the least norm solution of (3) is nondegenerate, all common zero indices of the error vectors $(Ax^0 - d)$ (with x^0 in the solution set of (1)) can be identified by the Huber M-estimator with a small tuning parameter γ and a linear ℓ_1 estimator can be explicitly constructed by solving another Huber M-estimator problem. However, when the least norm solution of (3) is degenerate, we have to repeatedly solve Huber M-estimator problems in order to find the common zero indices. Therefore, we propose a recursive version of Madsen and Nielsen's method for computing a linear ℓ_1 estimator by repeatedly solving Huber M-estimator problems. The new algorithm explicitly constructs a linear ℓ_1 estimator based on Huber M-estimators without Madsen and Nielsen's assumption that guarantees the uniqueness of both the Huber M-estimator and the linear ℓ_1 estimator.

REFERENCES

- [1] I. BARRODALE AND F. ROBERTS, *An improved algorithm for discrete L_1 linear approximation*, SIAM J. Numer. Anal., 10 (1972), pp. 839–848.
- [2] R. H. BARTELS, A. R. CONN, AND J. W. SINCLAIR, *Minimization techniques for piecewise differentiable functions: The L_1 -solution to an overdetermined linear system*, SIAM J. Numer. Anal., 15 (1978), pp. 224–240.
- [3] P. BLOOMFIELD AND W. L. STEIGER, *Least Absolute Deviations: Theory, Applications, and Algorithms*, Birkhäuser, Boston, MA, 1983.
- [4] J. V. BURKE AND M. C. FERRIS, *Weak sharp minima in mathematical programming*, SIAM J. Control Optim., 31 (1993), pp. 1340–1359.
- [5] D. I. CLARK, *The mathematical structure of Huber's M-estimator*, SIAM J. Sci. Statist. Comput., 6 (1985), pp. 209–219.
- [6] D. I. CLARK AND M. R. OSBORNE, *Finite algorithms for Huber's M-estimator*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 72–85.

- [7] H. EKBLOM, *A new algorithm for the Huber estimator in linear models*, BIT, 28 (1988), pp. 123–132.
- [8] M. C. FERRIS, *Finite termination of the proximal point algorithm*, Math. Programming, 50 (1991), pp. 359–366.
- [9] R. GONIN AND A. H. MONEY, *Nonlinear L_p -Norm Estimation*, Marcel Dekker, Inc., New York, 1989.
- [10] A. J. HOFFMAN, *On approximate solutions of systems of linear inequalities*, J. Res. Natl. Bur. Standards, 49 (1952), pp. 263–265.
- [11] P. J. HUBER, *Robust regression: Asymptotics, conjectures and Monte Carlo*, Ann. Statist., 1 (1973), pp. 799–821.
- [12] P. J. HUBER, *Robust Statistics*, John Wiley, New York, 1981.
- [13] W. LI, *Linearly convergent descent methods for unconstrained minimization of convex quadratic splines*, J. Optim. Theory Appl., 86 (1995), pp. 145–172.
- [14] W. LI, *A conjugate gradient method for the unconstrained minimization of strictly convex quadratic splines*, Math. Programming, 72 (1996), pp. 17–32.
- [15] W. LI, *Numerical algorithms for the Huber M-estimator problem*, in Approximation Theory VIII, C. K. Chui and L. Schumaker, eds., World Scientific Publishing Co., Inc., New York, 1995, pp. 325–334.
- [16] W. LI, *The sharp Lipschitz constants for feasible and optimal solutions of a perturbed linear program*, Linear Algebra Appl., 187 (1993), pp. 15–40.
- [17] W. LI, P. PARDALOS, AND C. G. HAN, *Gauss-Seidel method for least distance problems*, J. Optim. Theory Appl., 75 (1992), pp. 487–500.
- [18] K. MADSEN AND H. B. NIELSEN, *Finite algorithms for robust linear regression*, BIT, 30 (1990), pp. 682–699.
- [19] K. MADSEN AND H. B. NIELSEN, *A finite smoothing algorithm for linear ℓ_1 estimation*, SIAM J. Optim., 3 (1993), pp. 223–235.
- [20] K. MADSEN, H. B. NIELSEN, AND M. PINAR, *New characterizations of ℓ_1 solutions to overdetermined systems of linear equations*, Oper. Res. Lett., 16 (1994), pp. 159–166.
- [21] O. L. MANGASARIAN, *Normal solutions of linear programs*, Math. Programming Study, 22 (1984), pp. 206–216.
- [22] O. L. MANGASARIAN AND R. R. MEYER, *Nonlinear perturbations of linear programs*, SIAM J. Control Optim., 17 (1979), pp. 745–752.
- [23] O. L. MANGASARIAN AND T.-H. SHIAU, *Lipschitz continuity of solutions of linear inequalities, programs, and complementarity problems*, SIAM J. Control Optim., 25 (1987), pp. 583–595.
- [24] C. MICHELOT AND M. L. BOUGEARD, *Duality results and proximal solutions of the Huber M-estimator problem*, Appl. Math. Optim., 30 (1994), pp. 203–221.
- [25] K. G. MURTY, *Linear and Combinatorial Programming*, Wiley, New York, 1976.
- [26] M. R. OSBORNE, *Finite Algorithms in Optimization and Data Analysis*, John Wiley & Sons, New York, 1985.
- [27] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
- [28] S. A. RUZINSKY AND E. T. OLSEN, *L_1 and L_∞ minimization via a variant of Karmarkar's algorithm*, IEEE Trans. Acoustic Speech and Signal Processing, 37 (1989), pp. 245–253.
- [29] D. F. SHANNO AND D. M. ROCKE, *Numerical methods for robust regression: Linear models*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 86–97.
- [30] G. A. WATSON, *Approximation Theory and Numerical Methods*, Wiley, New York, 1980.
- [31] Y. ZHANG, *Primal-dual interior point approach for computing ℓ_1 -solutions and ℓ_∞ -solutions of overdetermined linear systems*, J. Optim. Theory Appl., 77 (1993), pp. 323–341.

A D.C. OPTIMIZATION ALGORITHM FOR SOLVING THE TRUST-REGION SUBPROBLEM*

PHAM DINH TAO[†] AND LE THI HOAI AN[†]

Abstract. This paper is devoted to difference of convex functions (d.c.) optimization: d.c. duality, local and global optimality conditions in d.c. programming, the d.c. algorithm (DCA), and its application to solving the trust-region problem. The DCA is an iterative method that is quite different from well-known related algorithms. Thanks to the particular structure of the trust-region problem, the DCA is very simple (requiring only matrix-vector products) and, in practice, converges to the global solution. The inexpensive implicitly restarted Lanczos method of Sorensen is used to check the optimality of solutions provided by the DCA. When a nonglobal solution is found, a simple numerical procedure is introduced both to find a feasible point having a smaller objective value and to restart the DCA at this point. It is shown that in the nonconvex case, the DCA converges to the global solution of the trust-region problem, using only matrix-vector products and requiring at most $2m+2$ restarts, where m is the number of distinct negative eigenvalues of the coefficient matrix that defines the problem. Numerical simulations establish the robustness and efficiency of the DCA compared to standard related methods, especially for large-scale problems.

Key words. d.c. optimization, d.c. duality, global and local optimality conditions, regularization techniques, DCA, Lanczos method, trust-region subproblem

AMS subject classifications. 65K05, 65K10, 90C30, 90C35

PII. S1052623494274313

1. Introduction. In this paper we shall be concerned with the following nonconvex quadratic optimization problems with nonempty solution sets:

$$(Q_1) \quad \min \left\{ \frac{1}{2} x^T A x + b^T x : \|x\| \leq r \right\},$$

$$(Q_2) \quad \min \left\{ \frac{1}{2} x^T A x + b^T x : \|x\| = r \right\},$$

where A is an $n \times n$ real symmetric matrix, $b \in \mathbb{R}$, r is a positive number and $\|\cdot\|$ denotes the Euclidean norm of $\mathbb{R}^n$. If A is positive semidefinite, then (Q_1) is a convex quadratic problem; in general, (Q_1) is nonconvex. Problem (Q_2) , whose feasible domain is a sphere, is always nonconvex even if A is positive semidefinite. (Q_1) and (Q_2) are among a few nonconvex optimization problems which possess a complete characterization of their solutions. These problems play an important role in optimization and numerical analysis. For example, for solving the problem of minimizing a twice continuously differentiable function on the whole space $\mathbb{R}^n$, D. M. Gay [5], J. J. Moré and D. C. Sorensen [16], [17], [36], and T. Pham Dinh and his coworkers [1], [7], [24], [25], [26] have used trust-region methods. It is known that these methods are of Newton type and consist of solving subproblems of the form (Q_1) ; consequently, (Q_1) has been often called a trust-region subproblem. Furthermore, in numerical analysis, several problems, e.g., constrained eigenvalue and quadratically

*Received by the editors August 20, 1994; accepted for publication (in revised form) November 6, 1996.

<http://www.siam.org/journals/siopt/8-2/27431.html>

[†]Mathematical Modelling and Applied Optimization Group, LMI-INSa Rouen - CNRS URA 1378, BP 08, 76131 Mont Saint Aignan, France (pham@insa-rouen.fr, lethi@insa-rouen.fr).

constrained least squares problems, can be formulated in the forms of (Q_1) and (Q_2) [4], [6].

To our knowledge, the first efficient (primal-dual) algorithm for solving (Q_1) is due to J. J. Moré and D. C. Sorensen [17]. These authors used a safeguarding technique combined with an adapted Newton method (due to Hebden) for computing a saddle-point of the Lagrangian function $L(x, \lambda)$. In the same primal-dual approach T. Pham Dinh and L. T. Hoai An [24], [26] proposed a dichotomy process as an alternative for the safeguarding technique of J. J. Moré and D. C. Sorensen. According to many comparative numerical simulations [7], [24], [26], the dichotomy algorithm seems to be less expensive than the safeguarding one. Moreover, it can be also applied to solving (Q_2) .

G. H. Golub, W. Gander, and U. Von Matt [4], [6] studied (Q_2) from both theoretical and computational viewpoints. These authors mentioned algorithmic difficulties of (Q_2) caused by the nonconvexity of the constraint $\{x : \|x\| = r\}$. In [4] the sensitivity of the solutions to the primal problem was discussed. More recently, G. H. Golub and U. Von Matt [6] proposed a new computational approach to the solution of (Q_2) in the normal case. The method again uses the theoretical tools developed in [4]. It seems to be well suited when A is a large sparse matrix.

J. M. Martinez [15] has investigated the nature of local nonglobal solutions of (Q_1) and (Q_2) and has shown the following interesting property (especially for local algorithms): these problems have at most one local nonglobal solution. Moreover, being inspired by G. E. Forsythe and G. H. Golub's work [3], S. Lucidi, L. Palagi, and M. Roma [13] have stated a very nice result: in (Q_1) the objective function can admit at most $2m + 2$ different values at Kuhn–Tucker points, where m is the number of distinct negative eigenvalues of A .

The algorithms in [17], [24] require matrix factorizations which are not feasible in the large-scale setting. Recently, matrix-free methods [32], [35], [38] have been introduced to solve large-scale trust-region subproblems. These methods recast the trust-region subproblem in terms of a parametrized eigenvalue problem and require matrix–vector products only. Sorensen's algorithm [38] provides a superlinearly convergent scheme to adjust the parameter and find the optimal solution from the eigenvector of the parametrized problem, as long as the hard case does not occur. The so-called hard case [17] is characterized by whenever the vector b is orthogonal to the eigenspace of A corresponding to its smallest eigenvalue λ_1 and there is a solution u to the system $(A - \lambda_1 I)u = -b$ such that $\|u\| < r$. This algorithm is linearly convergent in the hard case. The implicitly restarted Lanczos method (IRLM) introduced by D. C. Sorensen [12], [37] which meets the requirements of limited storage and reliance only on matrix–vector products is the recommended technique used in [38] to find the smallest eigenvalue and the corresponding eigenvector of the parameterized problem. A primal-dual semidefinite framework for the trust-region subproblem is proposed by F. Rendl and H. Wolkowicz in [32], where a dual simplex-type method is used in the general iteration and a primal simplex-type method provides steps for the hard case iteration. In [32] a block Lanczos routine is used at each iteration instead of the IRLM. The new matrix-free algorithm of S. A. Santos and D. C. Sorensen [35] improves upon that of D. C. Sorensen by introducing a unified iteration that naturally includes the hard case. It is shown that this algorithm is superlinearly convergent in all cases. Numerical simulations given in [35] proved the superiority of S. A. Santos and D. C. Sorensen's algorithm upon that of D. C. Sorensen. Moreover, these tests indicate that the former has advantage over F. Rendl and H. Wolkowicz's algorithm.

In convex approach to nonconvex nondifferentiable optimization the first author has extensively studied subgradient methods for solving convex maximization problems (see [20] and references therein). The d.c. duality investigated by J. F. Toland [39] and the works on d.c. optimization by T. Pham Dinh and L. T. Hoai An [7], [27], [28], [29], [30] can be considered as a natural generalization of the aforementioned works concerning convex maximization. DCA based on the duality in d.c. optimization had been introduced by T. Pham Dinh and S. Elberoussi in [23]. Important developments and improvements for the DCA from both theoretical and numerical viewpoints have been completed after the works by the authors [7], [8], [27], [28], [29], [30] appeared. These algorithms are actually among a few algorithms which allow to solve large-scale d.c. optimization problems [7], [8]. Due to their local character they cannot guarantee the globality of computed solutions for general d.c. programs. In general the DCAs, converge to a local solution; however, we observe that the DCAs, converge quite often to a global one [7], [8].

The d.c. objective function (of a d.c. program) has infinitely many d.c. decompositions which may have an important influence on the qualities (robustness, stability, rate of convergence, and globability of sought solutions) of the DCA. So, it is particularly interesting to obtain various equivalent d.c. forms for the primal and dual problems. The Lagrangian duality without gap in d.c. optimization [7], [28], [30] and regularization techniques partially answer this concern. In practice, regularization techniques using the kernel $\lambda/2\|\cdot\|^2$ and inf-convolution may provide interesting d.c. decompositions of objective functions for the DCA [7], [27], [29]. Furthermore, it is worth noting that by using conjointly suitable d.c. decompositions of convex functions and proximal regularization techniques [7], [27], [29] we can obtain the proximal point algorithm [14], [34] and the Goldstein–Levitin–Polyak subgradient projection method [31] as particular cases of the DCA. It would be interesting to find conditions on the choice of the d.c. decompositions and the initial point x^o to ensure the convergence of the DCA to a global minimizer. In practice, the DCA have been successfully applied to many large-scale d.c. optimization problems and proved to be more robust and efficient than related standard methods [7], [8], [27], [29].

The main part of the paper is dealing with the d.c. optimization framework (d.c. duality, local and global optimalities for d.c. programming, and the DCA) and its application to the treatment of problem (Q_1) since (Q_2) can be simply transformed into an equivalent problem of the form (Q_1) .

We first briefly present global optimality conditions in (Q_1) and (Q_2) [1], [2], [5], [7], [17], [24], [25], [26], [36]. These results express the same important fact: writing the constraints in (Q_1) and (Q_2) in equivalent forms, we obtain [26], [30] the stability of the Lagrangian duality relative to them, i.e., there is no duality gap and the solution sets of (Q_1) and (Q_2) can be described with the help of their dual solutions exactly as in convex programming. This approach emphasizes main properties of the dual objective function and is particularly useful to the development of various primal-dual algorithms for solving these problems. The algorithms in [17], [24], [32], [35], [38] are originated from this primal-dual approach.

In the third section we study the d.c. optimization framework, the description of the DCA and its convergence. The key point which makes a unified and deep d.c. optimization theory possible relies on the particular structure of the objective function to be minimized on $\mathbb{R}^n$:

$$f(x) = g(x) - h(x)$$

with g and h being convex on $\mathbb{R}^n$. One works then actually with the powerful convex

analysis tools applied to the two d.c. components g and h of the d.c. function f . With the help of the functional conjugate notion, the d.c. duality associates a primal d.c. program with a dual d.c. one and states relationships between them. More precisely, the d.c. duality is built on the fundamental feature of proper lower semicontinuous convex functions; such functions $\theta(x)$ are the supremum of the affine functions

$$\sup\{\langle x, y \rangle - \theta^*(y) : y \in \mathbb{R}^n\}.$$

Thanks to a symmetry in the d.c. duality (the bidual d.c. program is exactly the primal one) and the d.c. duality transportation of global minimizers, solving a d.c. program implies solving the dual one and vice versa. Furthermore, it may be very useful when one program is easier to be solved than the other (for example, one can directly solve Problem (Q_1) but not its dual d.c. program). The equality of the optimal value in the primal and dual d.c. programs can be easily translated (with the help of ε -subdifferentials of the d.c. components) in global optimality conditions. These conditions have been stated for the first time by J. B. Hiriart-Urruty [9] in a complicated way and mark the passage from convex optimization to nonconvex optimization. They are nice but difficult to use for devising solution methods to d.c. programs.

Local d.c. optimality conditions constitute the basis of the DCA. In general, it is not easy to state them as in global d.c. optimality and very few properties have been found which are useful in practice. Therefore, we shall present in section 3 the most significative results on d.c. programming; most of them are new. In particular, we give there a new elegant property concerning sufficient local d.c. optimality conditions and their consequences, e.g., the d.c. duality transportation of local minimizers ((ii) of Theorem 6). The latter is very helpful in establishing relationships between local minimizers of primal and dual d.c. programs. All these results are indispensable in the understanding of DCA for locally solving primal and dual d.c. programs. The proof of the convergence of the DCA will be presented in the appendix.

An application of the DCA to solving the trust-region subproblem (Q_1) is given in the fourth section. The DCA for this problem is quite different from the aforementioned algorithms: it is an iterative method based on the local optimality in d.c. programming. For solving (Q_1) we propose a quite appropriate d.c. decomposition. It turns out that the DCA becomes very simple: it indifferently treats both the standard (or normal) and hard cases and requires matrix-vector products only. Theoretically, the DCA generates a sequence of feasible points $\{x^k\}$ which decrease the values of the original quadratic function and converges to a Kuhn-Tucker point of (Q_1) . Main results for d.c. optimization show the convergence of the DCA to a local solution for certain classes of d.c. programs. These do not concern (Q_1) directly; however, the foundations of the d.c. optimization theory enable us to conjecture that the DCA converges to a local solution to (Q_1) and so, in general, to a global solution to this problem according to J. M. Martinez's results [15]. For the trust-region subproblem, it is interesting to note the following paradox: checking a global solution is easier than checking a local nonglobal one [15]. We take advantage of this fact to introduce a simple numerical procedure for checking the global optimality of solutions x^* given by the DCA or finding a feasible point $\bar{x}$ having a smaller objective value in case x^* is not global. This procedure uses the efficient and inexpensive IRLM to compute the smallest eigenvalue of A and a corresponding eigenvector. In case of nonglobal solutions, we restart DCA with the new initial point $\bar{x}$. Finally, we can affirm that the DCA (with at most $2m + 2$ restarting procedures), which requires only matrix-vector products too, converges to a global solution.

In the last section we present computational results and comments. A lot of numerical experiments have been done to compare the DCA with some known related algorithms, among them the new matrix-free algorithm of S. A. Santos and D. C. Sorensen [35]. They have proved the robustness and the efficiency of the DCA, especially in the large-scale setting, and the fact that in practice the DCA (without restarting procedure) converges to a global solution of the trust-region subproblem.

2. Global optimality conditions for (Q_1) and (Q_2) . We first summarize several well-known global optimality conditions for (Q_1) whose proofs can be found in [1], [16], [17], [25], [26], [30], [36].

THEOREM 2.1. *x^* is a solution to Problem (Q_1) if and only if there exists $\lambda^* \geq 0$ such that*

- (i) $(A + \lambda^* I)x^* = -b$,
- (ii) $\lambda^*(\|x^*\| - r) = 0, \|x^*\| \leq r$,
- (iii) $(A + \lambda^* I)$ is positive semidefinite.

Moreover, λ^* is unique.

The conditions (i) and (ii) are Kuhn–Tucker conditions (necessary local optimality conditions). So, only the last condition is crucial and does express the passage from “convex” to “nonconvex” in (Q_1) .

We note that (Q_2) is equivalent to a problem of the form (Q_1) . Namely, it is equivalent to

$$(1) \quad \min \left\{ \frac{1}{2} x^T (A + \gamma I) x + b^T x \quad : \quad \|x\| \leq r \right\},$$

where γ is a real number such that $A + \gamma I$ is not positive semidefinite. Indeed, (Q_2) is equivalent to

$$(2) \quad \min \left\{ \frac{1}{2} x^T (A + \gamma I) x + b^T x \quad : \quad \|x\| = r \right\},$$

and in case $A + \gamma I$ is not positive semidefinite problems (1) and (2), by Theorem 2.1, have the same sets of solutions. Thus, solving (Q_2) is reduced to solving problem (1) which is of the form (Q_1) .

In [36] D. C. Sorensen gave a sufficient condition for optimality to problem (Q_2) . In [24] T. Pham Dinh proved that this condition is not only sufficient, but also necessary. This result was also given partially in R. Fletcher [2] using the optimality conditions of Theorem 2.1. It looks as follows.

THEOREM 2.2. *x^* is a solution to (Q_2) if and only if there exists λ^* such that*

- (i) $(A + \lambda^* I)$ is positive semidefinite,
- (ii) $(A + \lambda^* I)x^* = -b$,
- (iii) $\|x^*\| = r$.

This number λ^* is unique.

Denote by $\lambda_1 \leq \dots \leq \lambda_n$ the eigenvalues of A . Let ϕ be a function defined on $\mathbb{R} \setminus \{-\lambda_i : i = 1, \dots, n\}$ by

$$\phi(\lambda) = \|x(\lambda)\|,$$

where $x(\lambda)$ is the solution of $(A + \lambda I)x = -b$.

The following nice result which has been stated very recently by J. Martinez [15] strengthens the ability for the DCA to reach a solution (global minimizer) of (Q_1) .

THEOREM 2.3. (i) If x^* is a local nonglobal minimizer for (Q_1) or (Q_2) , then $(A + \lambda^* I)x^* = -b$ with $\lambda^* \in]-\lambda_2, -\lambda_1[$ and $\phi'(\lambda^*) \geq 0$. If x^* is a local nonglobal minimizer for (Q_1) , then $\lambda^* \geq 0$.

(ii) There exists at most one local nonglobal minimizer for (Q_1) or (Q_2) .

(iii) If $\|x^*\| = r$, $(A + \lambda^* I)x^* = -b$ for some $\lambda^* \in]-\lambda_2, -\lambda_1[$ and $\phi'(\lambda^*) > 0$, then x^* is a strict local minimizer for (Q_2) . If, in addition, $\lambda^* > 0$, then x^* is also a strict local minimizer for (Q_1) .

(iv) If b is orthogonal to some eigenvector associated with λ_1 , then there is no local nonglobal minimizer for (Q_1) and (Q_2) .

Finally, a result in S. Lucidi, L. Palagi, and M. Roma [13] guarantees that in the nonconvex case the restarting procedure can be repeated at most $2m + 2$ times, where m is the number of the distinct negative eigenvalues of A .

THEOREM 2.4. In (Q_1) the objective function can admit at most $2m + 2$ different values at Kuhn–Tucker points, where m is the number of the distinct negative eigenvalues of A .

3. D.c. optimization. Let the space $X = \mathbb{R}^n$ be equipped with the canonical inner product $\langle \cdot, \cdot \rangle$. Thus the dual space Y of X can be identified with X itself. The Euclidean norm of X is denoted by $\|\cdot\| = \langle \cdot, \cdot \rangle^{1/2}$. Denote by $\Gamma_o(X)$ the set of all proper lower semicontinuous convex functions on X . The conjugate function g^* of $g \in \Gamma_o(X)$ is a function belonging to $\Gamma_o(Y)$ and defined by

$$g^*(y) = \sup\{\langle x, y \rangle - g(x) : x \in X\}.$$

For a convex set C in X the indicator function of C is denoted by $\chi_C(x) = 0$ if $x \in C$, $+\infty$ otherwise. We shall use the following usual notations of [33]

$$\text{dom } g = \{x \in X : g(x) < +\infty\}.$$

For $\epsilon > 0$ and $x^o \in \text{dom } g$, the symbol $\partial_\epsilon g(x^o)$ denotes ϵ -subdifferential of g at x^o , i.e.,

$$\partial_\epsilon g(x^o) = \{y \in Y : g(x) \geq g(x^o) + \langle x - x^o, y \rangle - \epsilon \quad \forall x \in X\},$$

while $\partial g(x^o)$ stands for the usual (or exact) subdifferential of g at x^o . Also, $\text{dom } \partial g = \{x \in X : \partial g(x) \neq \emptyset\}$ and $\text{range } \partial g = \cup\{\partial g(x) : x \in \text{dom } \partial g\}$. We adopt in the sequel the convention $+\infty - (+\infty) = +\infty$.

A d.c. program is that of the form

$$(P) \quad \alpha = \inf\{f(x) := g(x) - h(x) : x \in X\},$$

where g and h belong to $\Gamma_o(X)$. Such a function f is called a d.c. function on X while g and h are called its d.c. components.

Using the definition of conjugate functions, we have

$$\begin{aligned} \alpha &= \inf\{g(x) - h(x) : x \in X\} = \inf\{g(x) - \sup\{\langle x, y \rangle - h^*(y) : y \in Y\} : x \in X\} \\ &= \inf\{\beta(y) : y \in Y\} \end{aligned}$$

with

$$(P_y) \quad \beta(y) = \inf\{g(x) - (\langle x, y \rangle - h^*(y)) : x \in X\}.$$

It is clear that $\beta(y) = h^*(y) - g^*(y)$ if $y \in \text{dom } h^*$, $+\infty$ otherwise. Finally, we state the dual problem

$$\alpha = \inf\{h^*(y) - g^*(y) : y \in \text{dom } h^*\}$$

which is written according to the above convention as

$$(D) \quad \alpha = \inf\{h^*(y) - g^*(y) : y \in Y\}.$$

We observe the perfect symmetry between primal and dual programs (P) and (D): the dual program to (D) is exactly (P).

Note that the finiteness of α merely implies that

$$(3) \quad \text{dom } g \subset \text{dom } h \quad \text{and} \quad \text{dom } h^* \subset \text{dom } g^*.$$

Such inclusions will be assumed throughout the paper.

This d.c. duality was first studied by J. F. Toland [39] in a more general framework. It can be considered as a logical generalization of our earlier works concerning convex maximization ([20] and references therein).

It is worth noting the richness of the set of d.c. functions on X , denoted by $\mathcal{DC}(X)$ [7], [9], [10], [27]:

(i) $\mathcal{DC}(X)$ is a subspace containing the class of lower- $\mathcal{C}^2$ functions (f is said to be lower- $\mathcal{C}^2$ if f is locally a supremum of a family of $\mathcal{C}^2$ functions). In particular, $\mathcal{DC}(X)$ contains the space $\mathcal{C}^{1,1}(X)$ of functions whose gradient is locally Lipschitzian on X .

(ii) $\mathcal{DC}(X)$ is closed under all the operations usually considered in optimization. In particular, a linear combination of $f_i \in \mathcal{DC}(X)$ belongs to $\mathcal{DC}(X)$, a finite supremum of d.c. functions is d.c. Moreover, the product of two d.c. functions remains d.c. [7], [9], [10], [27].

(iii) Under some caution we can say that $\mathcal{DC}(X)$ is the subspace generated by the convex cone $\Gamma_o(X) : \mathcal{DC}(X) = \Gamma_o(X) - \Gamma_o(X)$. This relation marks the passage from convex optimization to nonconvex optimization and also indicates that $\mathcal{DC}(X)$ constitutes a minimal realistic extension of $\Gamma_o(X)$.

A point x^* is said to be a *local minimizer* of $g - h$ if $g(x^*) - h(x^*)$ is finite (i.e., $x^* \in \text{dom } g \cap \text{dom } h$) and there exists a neighborhood U of x^* such that

$$(4) \quad g(x^*) - h(x^*) \leq g(x) - h(x) \quad \forall x \in U.$$

Under the convention $+\infty - (+\infty) = +\infty$, the property (4) is equivalent to $g(x^*) - h(x^*) \leq g(x) - h(x)$, $\forall x \in U \cap \text{dom } g$.

A point x^* is said to be a *critical point* of $g - h$ if $\partial g(x^*) \cap \partial h(x^*) \neq \emptyset$.

The interior of the set S in X is denoted by $\text{int } S$. Moreover, if S is convex then $\text{ri } S$ stands for the relative interior of S .

A convex function f on X is said to be *essentially differentiable* if it satisfies the following three conditions [33]:

- (i) $C = \text{int } (\text{dom } f) \neq \emptyset$,
- (ii) f is differentiable on C ,
- (iii) $\lim_{k \rightarrow \infty} \|\nabla f(x^k)\| = +\infty$ for every sequence $\{x^k\}$ which converges to a point at the boundary of C .

Let $\rho \geq 0$ and C be a convex subset of X . One says that the function $\theta : C \rightarrow \mathbb{R} \cup \{+\infty\}$ is ρ -convex if

$$\theta[\lambda x + (1 - \lambda)x'] \leq \lambda\theta(x) + (1 - \lambda)\theta(x') - \frac{\lambda(1 - \lambda)}{2}\rho\|x - x'\|^2 \quad \forall \lambda \in]0, 1[\quad \forall x, x' \in C.$$

It amounts to say that $\theta - (\rho/2)\|\cdot\|^2$ is convex on C . The modulus of strong convexity of θ on C , denoted by $\rho(\theta, C)$ or $\rho(\theta)$ if $C = X$, is given by

$$(5) \quad \rho(\theta, C) = \sup\{\rho \geq 0 : \theta - (\rho/2)\|\cdot\|^2 \text{ is convex on } C\}.$$

Clearly, the ρ -convexity implies the convexity. One says that θ is *strongly convex* on C if $\rho(\theta, C) > 0$.

For $f \in \Gamma_o(X)$ and $\lambda > 0$ the Moreau–Yosida regularization of f with parameter λ , denoted by f_λ , is the inf-convolution of f and $1/2\lambda\|\cdot\|^2$. The function f_λ is continuously differentiable, underapproximates f without changing the set of minimizers and $(f_\lambda)_\mu = f_{\lambda+\mu}$. More precisely, $\nabla f_\lambda = 1/\lambda[I - (I + \lambda\partial f)^{-1}]$ is Lipschitzian with the ratio $1/\lambda$. The operator $(I + \lambda\partial f)^{-1}$ is called the proximal mapping associated to λf [34].

Let $\mathcal{P}$ and $\mathcal{D}$ denote the solution sets of problems (P) and (D), respectively, and let

$$\mathcal{P}_l = \{x^* \in X : \partial h(x^*) \subset \partial g(x^*)\}, \quad \mathcal{D}_l = \{y^* \in Y : \partial g^*(y^*) \subset \partial h^*(y^*)\}.$$

We present below some fundamental results of d.c. optimization which constitute the basis of the DCA presented in subsection 3.3.

3.1. Duality and global optimality for d.c. optimization.

THEOREM 3.1. *Let $\mathcal{P}$ and $\mathcal{D}$ be the solution sets of problems (P) and (D), respectively. Then*

- (i) $x \in \mathcal{P}$ if and only if $\partial_\epsilon h(x) \subset \partial_\epsilon g(x) \quad \forall \epsilon > 0$.
- (ii) Dually, $y \in \mathcal{D}$ if and only if $\partial_\epsilon g^*(y) \subset \partial_\epsilon h^*(y) \quad \forall \epsilon > 0$.
- (iii) $\cup\{\partial h(x) : x \in \mathcal{P}\} \subset \mathcal{D} \subset \text{dom } h^*$.

The first inclusion becomes equality if g^* is subdifferentiable in $\mathcal{D}$ (in particular, if $\mathcal{D} \subset \text{ri}(\text{dom } g^*)$ or if g^* is subdifferentiable in $\text{dom } h^*$). In this case $\mathcal{D} \subset (\text{dom } \partial g^* \cap \text{dom } \partial h^*)$.

- (iv) $\cup\{\partial g^*(y) : y \in \mathcal{D}\} \subset \mathcal{P} \subset \text{dom } g$.

The first inclusion becomes equality if h is subdifferentiable in $\mathcal{P}$ (in particular, if $\mathcal{P} \subset \text{ri}(\text{dom } h)$ or if h is subdifferentiable in $\text{dom } g$). In this case $\mathcal{P} \subset (\text{dom } \partial g \cap \text{dom } \partial h)$.

The relationship between primal and dual solutions: $\cup\{\partial h(x) : x \in \mathcal{P}\} \subset \mathcal{D}$ and $\cup\{\partial g^*(y) : y \in \mathcal{D}\} \subset \mathcal{P}$ is due to J. F. Toland [39] in the general context of the duality principle dealing with linear vector spaces in separating duality. A direct proof of the results (except Properties (i) and (ii)), based on the theory of subdifferential for convex functions is given in [7], [23], [27]. The properties (i) and (ii) have been first established by J. B. Hiriart-Urruty [9]. His proof (based on his earlier work concerning the behavior of the ϵ -directional derivative of a convex function as a function of the parameters ϵ) is quite complicated. The following proof of these properties is very simple and well suited to our d.c. duality framework [7], [23], [27]. In fact, they are nothing but a geometrical translation of the equality of the optimal value in the primal and dual d.c. programs (P) and (D).

Indeed, in virtue of the d.c. duality, $x^* \in \mathcal{P}$ if and only if $x^* \in \text{dom } g$ and

$$(6) \quad g(x^*) - h(x^*) \leq h^*(y) - g^*(y) \quad \forall y \in \text{dom } h^*,$$

i.e.,

$$(7) \quad g(x^*) + g^*(y) \leq h(x^*) + h^*(y) \quad \forall y \in \text{dom } h^*.$$

On the other hand, for $x^* \in \text{dom } h$, the property $\partial_\epsilon h(x^*) \subset \partial_\epsilon g(x^*) \quad \forall \epsilon > 0$ is, by definition, equivalent to

$$(8) \quad \forall \epsilon > 0, \langle x^*, y \rangle + \epsilon \geq h(x^*) + h^*(y) \Rightarrow \langle x^*, y \rangle + \epsilon \geq g(x^*) + g^*(y).$$

It is easy to see the equivalence between (7) and (8), and property (i) is thus proved. The global optimality condition in (i) is difficult to use for deriving solution methods to problem (P). The DCA which will be described in subsection 3.3 is based on local optimality conditions. The relations (ii) and (iv) indicate that solving the primal d.c. program (P) implies solving the dual d.c. program (D) and vice versa. It may be useful if one of them is easier to solve than the other.

3.2. Duality and local optimality conditions for d.c. optimization.

THEOREM 3.2. (i) *If x^* is a local minimizer of $g - h$, then $x^* \in \mathcal{P}_l$.*

(ii) *Let x^* be a critical point of $g - h$ and $y^* \in \partial g(x^*) \cap \partial h(x^*)$. Let U be a neighborhood of x^* such that $U \cap \text{dom } g \subset \text{dom } \partial h$. If for any $x \in U \cap \text{dom } g$ there is $y \in \partial h(x)$ such that $h^*(y) - g^*(y) \geq h^*(y^*) - g^*(y^*)$, then x^* is a local minimizer of $g - h$. More precisely,*

$$g(x) - h(x) \geq g(x^*) - h(x^*) \quad \forall x \in U \cap \text{dom } g.$$

Property (i) is well known [7], [9], [23], [39]. For completeness we give below a short proof for it. Property (ii) is new; it establishes an interesting sufficient condition (dealing with the d.c. duality) for local d.c. optimality.

Proof. (i) If x^* is a local minimizer of $g - h$, then there exists a neighborhood of x^* such that

$$g(x) - g(x^*) \geq h(x) - h(x^*) \quad \forall x \in U \cap \text{dom } g.$$

Hence for $y^* \in \partial h(x^*)$ we have $g(x) - g(x^*) \geq \langle x - x^*, y^* \rangle \quad \forall x \in U \cap \text{dom } g$. The convexity of g then implies that $y^* \in \partial g(x^*)$.

(ii) The condition $y^* \in \partial g(x^*) \cap \partial h(x^*)$ implies $g(x^*) + g^*(y^*) = \langle x^*, y^* \rangle = h(x^*) + h^*(y^*)$. Hence

$$(9) \quad g(x^*) - h(x^*) = h^*(y^*) - g^*(y^*).$$

For any $x \in U \cap \text{dom } g$, by assumption, there is $y \in \partial h(x)$ such that

$$(10) \quad h^*(y) - g^*(y) \geq h^*(y^*) - g^*(y^*).$$

On the other hand, we have $h(x) + h^*(y) = \langle x, y \rangle \leq g(x) + g^*(y)$. Hence,

$$(11) \quad g(x) - h(x) \geq h^*(y) - g^*(y).$$

Combining (9), (10), (11), we get $g(x) - h(x) \geq g(x^*) - h(x^*) \quad \forall x \in U \cap \text{dom } g$. $\square$

A function $\theta \in \Gamma_o(X)$ is said to be polyhedral convex if [33]

$$\theta(x) = \max\{\langle a_i, x \rangle - \alpha_i : i = 1, \dots, m\} + \chi_C(x) \quad \forall x \in X,$$

where C is a nonempty polyhedral convex set in X .

Polyhedral d.c. optimization occurs when either g or h is polyhedral convex. This class of d.c. optimization problems, which is frequently encountered in practice, enjoys interesting properties (from both theoretical and practical viewpoints) concerning

local optimality and the convergence of the DCA, and has been extensively developed in [7], [29].

COROLLARY 3.3 (sufficient local optimality). *Let x^* be a point that admits a neighborhood U such that $\partial h(x) \cap \partial g(x^*) \neq \emptyset \quad \forall x \in U \cap \text{dom } g$. Then x^* is a local minimizer of $g - h$. More precisely, $g(x) - h(x) \geq g(x^*) - h(x^*) \quad \forall x \in U \cap \text{dom } g$.*

Proof. Let $x \in U \cap \text{dom } g$ and let $y \in \partial h(x) \cap \partial g(x^*)$. Since $y \in \partial h(x)$ we have $h(x) + h^*(y) = \langle x, y \rangle \leq g(x) + g^*(y)$. So $g(x) - h(x) \geq h^*(y) - g^*(y)$. Similarly, $y \in \partial g(x^*)$ implies that $g(x^*) + g^*(y) = \langle x^*, y \rangle \leq h(x^*) + h^*(y)$. Hence $h^*(y) - g^*(y) \geq g(x^*) - h(x^*)$. If $y^* \in \partial h(x^*) \cap \partial g(x^*)$, then $g(x^*) + g^*(y^*) = \langle x^*, y^* \rangle = h(x^*) + h^*(y^*)$. Hence $g(x^*) - h(x^*) = h^*(y^*) - g^*(y^*)$. The assumptions of Theorem 3.2 (ii) are fulfilled. Thus, the proof is complete. $\square$

Corollary 3.3 can be used to prove the following well-known result due to C. Michelot (in the case where g, h belonging to $\Gamma_o(X)$ are finite on the whole X) and generalized by the authors [7], [29] to the case of arbitrary g and h belonging to $\Gamma_o(X)$: the converse of property (i) of Theorem 3.2 in case h is polyhedral convex.

COROLLARY 3.4 (sufficient strict local optimality). *If $x^* \in \text{int}(\text{dom } h)$ verifies $\partial h(x^*) \subset \text{int}(\partial g(x^*))$, then x^* is a strict local minimizer of $g - h$.*

Proof. From the upper semicontinuity of the operator ∂h at $x^* \in \text{int}(\text{dom } h)$ [33] it follows that for any open set O containing $\partial h(x^*)$ there is a neighborhood U of x^* such that $\partial h(x) \subset O \quad \forall x \in U$.

Hence by letting $O = \text{int}(\partial g(x^*))$ and taking Corollary 3.3 into account, we have x^* is a local minimizer of $g - h$. But x^* is actually a strict local minimizer of $g - h$. Indeed, since $\partial h(x)$ is compact for $x \in V = U \cap \text{int}(\text{dom } h)$, we have $\forall x \in V \quad \exists \epsilon(x) > 0$ such that $\partial h(x) + \epsilon(x)B \subset O$ (B being the closed unit ball of the Euclidean norm).

Now let $x \in V \setminus \{x^*\}$ and $y \in \partial h(x)$. Then

$$\begin{aligned} g(x) - g(x^*) &\geq \left\langle x - x^*, y + \frac{\epsilon(x)}{\|x - x^*\|} (x - x^*) \right\rangle \\ &= \epsilon(x) \|x - x^*\| + \langle x - x^*, y \rangle \\ &\geq \epsilon(x) \|x - x^*\| + h(x) - h(x^*). \end{aligned}$$

The proof is complete. $\square$

It may happen that it is easier to solve locally the dual d.c. program (D) than the primal d.c. program (P). So it is useful to state results relative to the d.c. duality transportation of local minimizers. Paradoxically, such a result is more complicated than the d.c. duality transportation of global minimizers in Theorem 3.1.

COROLLARY 3.5 (d.c. duality transportation of a local minimizer). *Let $x^* \in \text{dom } \partial h$ be a local minimizer of $g - h$, and let $y^* \in \partial h(x^*)$ (i.e., $\partial h(x^*)$ is nonempty and x^* admit a neighborhood U such that $g(x) - h(x) \geq g(x^*) - h(x^*) \quad \forall x \in U \cap \text{dom } g$). If*

$$(12) \quad y^* \in \text{int}(\text{dom } g^*) \quad \text{and} \quad \partial g^*(y^*) \subset U$$

((12) holds if g^* is differentiable at y^*), then y^* is a local minimizer of $h^* - g^*$.

Proof. According to (i) of Theorem 3.2 we have $y^* \in \partial h(x^*) \subset \partial g(x^*)$. So $x^* \in \partial g^*(y^*) \cap \partial h^*(y^*)$. Under the assumption (12) and the upper semicontinuity of ∂g^* , y^* admits a neighborhood $V \subset \text{int}(\text{dom } g^*)$ such that ([33]) $\partial g^*(V) \subset U$. More precisely, $\partial g^*(V) \subset (U \cap \text{dom } g)$ since we have [33] $\text{range } \partial g^* = \text{dom } \partial g$ and $\text{dom } \partial g \subset \text{dom } g$. Using the dual property (in the d.c. duality) in (ii) of Theorem 3.2

we deduce that y^* is a local minimizer of $h^* - g^*$. If g^* is differentiable at y^* , then $x^* = \partial g^*(y^*)$ and we have (12) [33]. $\square$

By the symmetry of the d.c. duality, Corollary 3.5 has its corresponding dual part.

Remark. This result improves an earlier result of J. F. Toland [39] where he assumed that g^* is differentiable on the whole dual space Y . In [7], [29] we have proved that this result remains true if g^* is only essentially differentiable.

3.3. Description of the DCA for general d.c. programs. For each fixed $x^* \in X$ we consider the problem

$$(S(x^*)) \quad \inf\{h^*(y) - g^*(y) : y \in \partial h(x^*)\},$$

which is equivalent to the convex maximization one

$$\inf\{\langle x^*, y \rangle - g^*(y) : y \in \partial h(x^*)\}.$$

Similarly, for each fixed $y^* \in Y$, for duality, we define the problem

$$(T(y^*)) \quad \inf\{g(x) - h(x) : x \in \partial g^*(y^*)\}.$$

This problem is equivalent to

$$\inf\{\langle x, y^* \rangle - h(x) : x \in \partial g^*(y^*)\}.$$

Let $S(x^*)$, $T(y^*)$ denote the solution sets of Problems $(S(x^*))$ and $(T(y^*))$, respectively.

The complete form of the DCA is based upon duality of d.c. optimization defined by (P) and (D). It allows approximating a point $(x^*, y^*) \in \mathcal{P}_l \times \mathcal{D}_l$. Given a point $x^o \in \text{dom } g$, the algorithm constructs two sequences $\{x^k\}$ and $\{y^k\}$ defined by

$$(13) \quad y^k \in S(x^k); \quad x^{k+1} \in T(y^k).$$

The complete DCA can be viewed as a sort of a decomposition approach of the primal and dual problems (P), (D). From a practical point of view, although problems $(S(x^k))$ and $(T(y^k))$ are simpler than (P), (D) (we work in $\partial h(x^k)$ and $\partial g^*(y^k)$ with convex maximization problems), they remain nonconvex programs and thus are still difficult (see subsection 3.5). In practice the following simplified form of the DCA is used:

• **Simplified form of the DCA:** The idea of the simplified DCA is quite natural: it constructs two sequences $\{x^k\}$ and $\{y^k\}$ (candidates to primal and dual solutions) which are easy to calculate and satisfy the following conditions:

- (i) The sequences $(g - h)(x^k)$ and $(h^* - g^*)(y^k)$ are decreasing.
- (ii) Every limit point x^* (resp. y^*) of the sequence $\{x^k\}$ (resp. $\{y^k\}$) is a critical point of $g - h$ (resp. $h^* - g^*$).

These conditions suggest constructing two sequences $\{x^k\}$ and $\{y^k\}$, starting from a given point $x^o \in \text{dom } g$, by setting

$$y^k \in \partial h(x^k); \quad x^{k+1} \in \partial g^*(y^k).$$

Interpretation of the simplified DCA: At each iteration k we do the following:

$$\begin{aligned} x^k &\in \partial g^*(y^{k-1}) \rightarrow y^k \in \partial h(x^k) \\ &= \operatorname{argmin}\{h^*(y) - [g^*(y^{k-1}) + \langle x^k, y - y^{k-1} \rangle] : y \in Y\}, \quad (D_k) \end{aligned}$$

$$y^k \in \partial h(x^k) \rightarrow x^{k+1} \in \partial g^*(y^k) = \operatorname{argmin}\{g(x) - [h(x^k) + \langle x - x^k, y^k \rangle] : x \in X\}. \quad (P_k)$$

Problem (P_k) is a convex program obtained from (P) by replacing h with its affine minorization defined by $y^k \in \partial h(x^k)$. Similarly, the convex problem (D_k) is obtained from (D) by using the affine minorization of g^* defined by $x^k \in \partial g^*(y^{k-1})$. Here we can see a complete symmetry between problems (P_k) and (D_k) , and between the sequences $\{x^k\}$ and $\{y^k\}$ relative to the duality of d.c. optimization. The two forms of the DCA are identical if g^* and h are essentially differentiable.

• **Well definiteness of the DCA:** The DCA is well defined if one can construct two sequences $\{x^k\}$ and $\{y^k\}$ as above from an arbitrary initial point $x^o \in \operatorname{dom} g$. We have $x^{k+1} \in \partial g^*(y^k)$ and $y^k \in \partial h(x^k) \forall k \geq 0$. So $\{x^k\} \subset \operatorname{range} \partial g^* = \operatorname{dom} \partial g$ and $\{y^k\} \subset \operatorname{range} \partial h = \operatorname{dom} \partial h^*$. Then the following lemma is clear.

LEMMA 3.6. *Sequences $\{x^k\}$, $\{y^k\}$ in the DCA are well defined if and only if*

$$\operatorname{dom} \partial g \subset \operatorname{dom} \partial h \text{ and } \operatorname{dom} \partial h^* \subset \operatorname{dom} \partial g^*.$$

Since for $\varphi \in \Gamma_o(X)$ we have $\operatorname{ri}(\operatorname{dom} \varphi) \subset \operatorname{dom} \partial \varphi \subset \operatorname{dom} \varphi$ ([33]) ($\operatorname{ri}(\operatorname{dom} \varphi)$ stands for the relative interior of $\operatorname{dom} \varphi$) we can say, under the essential assumption (3), that the DCA is, in general, well defined.

Remark. A d.c. function f has infinitely many d.c. decompositions. For example, if $f = g - h$, then $f = (g + \theta) - (h + \theta)$ for every $\theta \in \Gamma_o(X)$ finite on the whole X . It is clear that the primal d.c. programs (P) corresponding to the two d.c. decompositions of the objective function f are identical. But their dual programs are quite different and so is the DCA relative to these d.c. decompositions. In other words, there are as many DCAs as there are d.c. decompositions of the objective function f . It is useful to find a suitable d.c. decomposition of f since it may have an important influence on the efficiency of the DCA for its solution. This question is intimately related to the regularization techniques in d.c. programming [7], [23], [27].

3.4. Convergence of the DCA for general d.c. programs. Let ρ_i and ρ_i^* , ($i = 1, 2$) be real nonnegative numbers such that $0 \leq \rho_i < \rho(f_i)$ (resp. $0 \leq \rho_i^* < \rho(f_i^*)$) where $\rho_i = 0$ (resp. $\rho_i^* = 0$) if $\rho(f_i) = 0$ (resp. $\rho(f_i^*) = 0$) and ρ_i (resp. ρ_i^*) may take the value $\rho(f_i)$ (resp. $\rho(f_i^*)$) if it is attained. We next set $f_1 = g$ and $f_2 = h$. Also let $dx^k := x^{k+1} - x^k$ and $dy^k := y^{k+1} - y^k$.

The basic convergence theorem of the DCA for general d.c. programming will be stated below. Its proof is very technical and long. We will present it in the appendix and refer the reader to our previous works [7], [27] for more details.

THEOREM 3.7. *Suppose that the sequences $\{x^k\}$ and $\{y^k\}$ are defined by the simplified DCA. Then we have*

$$(i) \quad (g - h)(x^{k+1}) \leq (h^* - g^*)(y^k) - \max \left\{ \frac{\rho_2}{2} \|dx^k\|^2, \frac{\rho_2^*}{2} \|dy^k\|^2 \right\} \leq (g - h)(x^k) \\ - \max \left\{ \frac{\rho_1 + \rho_2}{2} \|dx^k\|^2, \frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2}{2} \|dx^k\|^2, \frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2^*}{2} \|dy^k\|^2 \right\}.$$

The equality $(g - h)(x^{k+1}) = (g - h)(x^k)$ holds if and only if $x^k \in \partial g^*(y^k)$, $y^k \in \partial h(x^{k+1})$ and $(\rho_1 + \rho_2)dx^k = \rho_1^*dy^{k-1} = \rho_2^*dy^k = 0$. In this case

• $(g - h)(x^{k+1}) = (h^* - g^*)(y^k)$ and x^k, x^{k+1} are the critical points of $g - h$ satisfying $y^k \in (\partial g(x^k) \cap \partial h(x^k))$ and $y^k \in (\partial g(x^{k+1}) \cap \partial h(x^{k+1}))$,

- y^k is a critical point of $h^* - g^*$ satisfying $[x^k, x^{k+1}] \subset ((\partial g^*(y^k) \cap \partial h^*(y^k)))$,
 - $x^{k+1} = x^k$ if $\rho(g) + \rho(h) > 0$, $y^k = y^{k-1}$ if $\rho(g^*) > 0$ and $y^k = y^{k+1}$ if $\rho(h^*) > 0$.
- (ii) Similarly, for the dual problem we have

$$(h^* - g^*)(y^{k+1}) \leq (g - h)(x^{k+1}) - \max \left\{ \frac{\rho_1}{2} \|dx^{k+1}\|^2, \frac{\rho_1^*}{2} \|dy^k\|^2 \right\} \leq (h^* - g^*)(y^k) \\ - \max \left\{ \frac{\rho_1}{2} \|dx^{k+1}\|^2 + \frac{\rho_2}{2} \|dx^k\|^2, \frac{\rho_1^*}{2} \|dy^k\|^2 + \frac{\rho_2}{2} \|dx^k\|^2, \frac{\rho_1^* + \rho_2^*}{2} \|dy^k\|^2 \right\}.$$

The equality $(h^* - g^*)(y^{k+1}) = (h^* - g^*)(y^k)$ holds if and only if $x^{k+1} \in \partial g^*(y^{k+1})$, $y^k \in \partial h(x^{k+1})$ and $(\rho_1^* + \rho_2^*)dy^k = \rho_2 dx^k = \rho_1 dx^{k+1} = 0$. In this case

- $(h^* - g^*)(y^{k+1}) = (g - h)(x^{k+1})$ and y^k, y^{k+1} are the critical points of $h^* - g^*$ satisfying $x^{k+1} \in (\partial g^*(y^k) \cap \partial h^*(y^k))$ and $x^{k+1} \in (\partial g^*(y^{k+1}) \cap \partial h^*(y^{k+1}))$,
- x^{k+1} is a critical point of $g - h$ satisfying $[y^k, y^{k+1}] \subset ((\partial g(x^{k+1}) \cap \partial h(x^{k+1})))$,
- $y^{k+1} = y^k$ if $\rho(g^*) + \rho(h^*) > 0$, $x^{k+1} = x^k$ if $\rho(h) > 0$ and $x^{k+1} = x^{k+2}$ if $\rho(g) > 0$.

(iii) If α is finite, then the decreasing sequences $\{(g - h)(x^k)\}$ and $\{(h^* - g^*)(y^k)\}$ converge to the same limit $\beta \geq \alpha$, i.e., $\lim_{k \rightarrow +\infty} (g - h)(x^k) = \lim_{k \rightarrow +\infty} (h^* - g^*)(y^k) = \beta$. If $\rho(g) + \rho(h) > 0$ (resp. $\rho(g^*) + \rho(h^*) > 0$), then $\lim_{k \rightarrow +\infty} \{x^{k+1} - x^k\} = 0$ (resp. $\lim_{k \rightarrow +\infty} \{y^{k+1} - y^k\} = 0$). Moreover, $\lim_{k \rightarrow +\infty} \{g(x^k) + g^*(y^k) - \langle x^k, y^k \rangle\} = 0 = \lim_{k \rightarrow +\infty} \{h(x^{k+1}) + h^*(y^k) - \langle x^{k+1}, y^k \rangle\}$.

(iv) If α is finite and the sequences $\{x^k\}$ and $\{y^k\}$ are bounded, then for every limit x^* of $\{x^k\}$ (resp. y^* of $\{y^k\}$) there exists a cluster point y^* of $\{y^k\}$ (resp. x^* of $\{x^k\}$) such that

- $(x^*, y^*) \in [\partial g^*(y^*) \cap \partial h^*(y^*)] \times [\partial g(x^*) \cap \partial h(x^*)]$ and $(g - h)(x^*) = (h^* - g^*)(y^*) = \beta$,
- $\lim_{k \rightarrow +\infty} \{g(x^k) + g^*(y^k)\} = \lim_{k \rightarrow +\infty} \langle x^k, y^k \rangle$.

Comments on Theorem 3.7. (i) Properties (i) and (ii) prove that the DCA is a descent method for both primal and dual programs. The DCA provides critical points for (P) and (D) after finitely many operations if there is no strict decrease of the primal (or dual) objective function.

(ii) If C and D are convex sets such that $\{x^k\} \subset C$ and $\{y^k\} \subset D$, then Theorem 3.7 remains valid if we replace $\rho(f_i)$ by $\rho(f_i, C)$ and $\rho(f_i^*)$ by $\rho(f_i^*, D)$ for $i = 1, 2$. By this way we may improve the results in the theorem.

(iii) In (ii) of Theorem 3.7, the convergence of the whole sequence $\{x^k\}$ (resp. $\{y^k\}$) can be ensured under the following conditions [18], [20]:

- $\{x^k\}$ is bounded;
- The set of limit points of $\{x^k\}$ is finite;
- $\lim_{k \rightarrow +\infty} \|x^{k+1} - x^k\| = 0$.

(iv) The only difference between the simplified DCA and the complete DCA lies on the choice of y^k in $\partial h(x^k)$ and x^{k+1} in $\partial g^*(y^k)$. The convergence result of the complete DCA is thus improved: in Theorem 3.7, the nonemptiness of a subdifferential intersection is replaced by a subdifferential inclusion [7], [23], [27]. In other words, the complete DCA permits to obtain a couple of elements $(x^*, y^*) \in \mathcal{P}_l \times \mathcal{D}_l$. In practice the simplified DCA usually yields a local minimizer which is also global [7].

(v) In general, the qualities (robustness, stability, rate of convergence, and globality of sought solutions) of the DCA, in both complete and simplified forms, depend upon the d.c. decomposition of the function f . Theorem 3.7 shows how strong convexity of d.c. components in primal and dual problems can influence the DCA. To

make the d.c. components (of the primal objective function $f = g - h$) strongly convex we usually apply the following process

$$f = g - h = \left(g + \frac{\lambda}{2} \|\cdot\|^2 \right) - \left(h + \frac{\lambda}{2} \|\cdot\|^2 \right).$$

In this case the d.c. components in the dual problem will be continuously differentiable. Similarly inf-convolution of g and h with $\lambda/2 \|\cdot\|^2$ will make the d.c. components (in dual problems) strongly convex and the d.c. components of the primal objective function continuously differentiable. For a detailed study of regularization techniques in the d.c. optimization, see [7], [23], [27].

3.5. How to restart the simplified DCA for obtaining x^* such that $\partial h(x^*) \subset \partial g(x^*)$. As mentioned above, the complete DCA theoretically provides a x^* such that $\partial h(x^*) \subset \partial g(x^*)$. In practice, except for the cases where the convex maximization problems $(S(x^k))$ and $(T(y^k))$ are easy to solve, one generally uses the simplified DCA. It is worth noting that if the simplified DCA terminates at some point x^* for which $\partial h(x^*)$ is not contained in $\partial g(x^*)$, then one can reduce the objective function value by restarting it from a new initial point $x^o = x^*$ with $y^o \in \partial h(x^o)$ such that $y^o \notin \partial g(x^o)$. In fact, since

$$g(x^1) + g^*(y^o) = \langle x^1, y^o \rangle \leq h(x^1) - h(x^o) + \langle x^o, y^o \rangle$$

and $\langle x^o, y^o \rangle < g(x^o) + g^*(y^o)$ because $y^o \notin \partial g(x^o)$, we have

$$g(x^1) + g^*(y^o) < h(x^1) - h(x^o) + g(x^o) + g^*(y^o).$$

Hence,

$$g(x^1) - h(x^1) < g(x^o) - h(x^o).$$

We have given the main concept of the d.c. programming and the DCA. In the next section we shall apply this technique to the trust-region subproblem.

4. Solving trust-region subproblem (Q_1) by the DCA.

4.1. DCA for solving Problem (Q_1) . We describe below the DCA for solving (Q_1) . For this we must point out d.c. decompositions of the objective function in (Q_1) .

$$(Q_1) \quad \alpha = \inf \left\{ \frac{1}{2} x^T A x + b^T x + \chi_E(x) : x \in \mathbb{R}^n \right\},$$

where $E = \{x \in \mathbb{R}^n : \|x\| \leq r\}$ and χ_E stands for its indicator function. The following d.c. decomposition seems to be the most natural

$$f(x) = \frac{1}{2} x^T A x + b^T x + \chi_E(x) = g_1(x) - h_1(x)$$

with

$$(14) \quad g_1(x) = \frac{1}{2} x^T (A + \rho I) x + x^T b + \chi_E(x), \quad h_1(x) = \frac{1}{2} \rho x^T x,$$

and ρ is a positive real number such that $A + \rho I$ is positive semidefinite.

For this case in the simplified DCA $y^k = \rho x^k$ and x^{k+1} is a solution of the problem

$$(15) \quad \min \left\{ \frac{1}{2} x^T (A + \rho I) x + x^T (b - y^k) : \|x\| \leq r \right\}.$$

Thus, at each iteration, the algorithm requires solving a convex quadratic program and so is expensive, especially in large-scale problems. In practice the algorithm with this d.c. decomposition is unstable ([7]). Note that if A is positive semidefinite then DCA with this decomposition reduces to the well-known proximal point algorithm [14], [34].

From the computational viewpoint the following decomposition, in our opinion, is the most efficient: $f(x) = g(x) - h(x)$ with

$$(16) \quad g(x) = \frac{1}{2} \rho \|x\|^2 + b^T x + \chi_E(x), \quad h(x) = \frac{1}{2} x^T (\rho I - A) x,$$

and ρ is a positive number such that $\rho I - A$ is positive semidefinite.

Clearly, $g, h \in \Gamma_o(\mathbb{R}^n)$ and thus (Q_1) takes the form

$$(17) \quad (P) \quad \min \{g(x) - h(x) : x \in \mathbb{R}\}.$$

The simplified DCA applied to this case can be formulated as follows. Let $x^o \in \mathbb{R}^n$ and $k \geq 0$. Further, let $y^k = (\rho I - A)x^k$ and x^{k+1} be a solution of the problem

$$(18) \quad \min \left\{ \frac{\rho}{2} \|x\|^2 + x^T (b - y^k) + \chi_E(x) : x \in \mathbb{R}^n \right\}.$$

Clearly,

$$(18) \Leftrightarrow \min_{x \in E} \left\| x - \frac{y^k - b}{\rho} \right\|^2.$$

Thus, x^{k+1} is, in fact, the projection of $(y^k - b)/\rho$ onto E , i.e.,

$$(19) \quad x^{k+1} = P_E \left(x^k - \frac{1}{\rho} (Ax^k + b) \right).$$

Summing up, we state the chosen DCA as follows:

DCA :

Let $x^o \in \mathbb{R}^n$ and $k \geq 0$

a) If $\|(\rho I - A)x^k - b\| \leq \rho r$ take $x^{k+1} = \frac{(\rho I - A)x^k - b}{\rho}$.

b) Otherwise, take $x^{k+1} = r \frac{(\rho I - A)x^k - b}{\|(\rho I - A)x^k - b\|}$.

If $\|x^{k+1} - x^k\| \leq \epsilon$, terminate. Otherwise, increase k by 1 and return to a).

Remark. (i) In case $b = 0$, except if A is positive semidefinite (for which zero is a solution), we choose x^o such that $y^o = (\rho I - A)x^o \neq 0$, since $y^o = 0$ implies $x^k = 0 \forall k \geq 1$.

(ii) We have presented the two d.c. decompositions (14), (16) (of the objective function f) which give rise to quite different DCA. These algorithms treat indifferently the normal and hard cases of (Q_1) . However, the DCA corresponding to the decomposition (16) requires only matrix-vector products since the solution (19) is explicit. This DCA seems to be more stable and more efficient than the former. The following convergence results are a consequence of Theorem 3.7.

THEOREM 4.1. (i) $f(x^{k+1}) \leq f(x^k) - 1/2(\rho + \lambda)\|x^{k+1} - x^k\|^2$, where λ is the smallest eigenvalue of $(\rho I - A)$.

(ii) $f(x^k) \searrow \alpha' \geq \alpha$ and $\lim_{k \rightarrow \infty} \|x^{k+1} - x^k\| = 0$.

(iii) Every limit point x^* of the sequence $\{x^k\}$ is a Kuhn-Tucker point; i.e., there exists $\lambda^* \geq 0$ such that $(A + \lambda^* I)x^* = -b$, $\lambda^*(\|x^*\| - r) = 0$ and $\|x^*\| \leq r$.

Proof. For the d.c. decomposition (16) of f , we have $\rho(g, E) = \rho$, $\rho(h) = \rho - \lambda_n$. Since the sequence $\{x^k\}$ is contained in E and the supremum $\rho(g, E)$ in (5) is attained, according to (i) of Theorem 3.7 and (i) of comments on Theorem 3.7, we get

$$f(x^{k+1}) \leq f(x^k) - \frac{1}{2}(2\rho - \lambda_n)\|x^{k+1} - x^k\|^2.$$

Property (ii) then follows from (iii) of the above theorem.

Now let x^* be a limit point of $\{x^k\}$. Since the sequences $\{x^k\}$ and $\{y^k\}$ are bounded, Property (iv) of Theorem 3.7 then implies

$$(20) \quad \partial h(x^*) = \nabla h(x^*) \subset \partial g(x^*).$$

We have $\nabla h(x^*) = (\rho I - A)x^*$, $\partial g(x^*) = \rho x^* + b + \partial \chi_E(x^*)$ with $\partial \chi_E(x^*) = \{0\}$ if $\|x^*\| < r$, $\{\lambda x^* : \lambda \geq 0\}$, otherwise. So the relation (20) is equivalent to

$$(A + \lambda^* I)x^* = -b, \lambda^*(\|x^*\| - r) = 0, \text{ and } \|x^*\| \leq r$$

for some nonnegative real λ^* . The proof is complete. $\square$

4.2. Using the DCA for globally solving (Q_1) . Let x^* be a point produced by the DCA (Theorem 4.1). It is clear that if A is positive semidefinite, then (P) in (17) is a “false” d.c. optimization problem (i.e., it is a convex one in reality) and the condition (iii) is actually a characterization of (Q_1) . In other words, the DCA converges to a solution of (Q_1) in this case. In general, using Theorem 2.1 we can calculate λ^* and check the global optimality. Indeed, from Theorem 2.1 it follows that

$$\lambda^* = \frac{-\langle x^*, Ax^* \rangle - \langle x^*, b \rangle}{r^2}.$$

So, checking the global optimality of x^* can be done, for example, by applying the inexpensive algorithm IRLM to compute the smallest eigenvalue λ_1 of A and a corresponding eigenvector u :

$$Au = \lambda_1 u.$$

If $\lambda^* + \lambda_1 \geq 0$, then x^* is a global solution to (Q_1) . Otherwise,

$$(21) \quad u^T(A + \lambda^* I)u = (\lambda^* + \lambda_1)u^T u < 0.$$

In this case we will indicate below how to restart the DCA, say, how to find $\bar{x}$ such that $\|\bar{x}\| \leq r$ and $f(\bar{x}) < f(x^*)$. First let us mention the following property.

LEMMA 4.2. If $f(x) = 1/2\langle x, Ax \rangle + \langle b, x \rangle$, with A being symmetric, and $(A + \mu I)y = -b$, then

$$f(x) = f(y) - \frac{\mu}{2}(\|x\|^2 - \|y\|^2) + \frac{1}{2}\langle x - y, (A + \mu I)(x - y) \rangle \quad \forall x.$$

Proof. We have

$$\begin{aligned} f(x) &= f(y) + \langle x - y, Ay + b \rangle + \frac{1}{2}\langle x - y, A(x - y) \rangle \\ &= f(y) - \mu\langle x - y, y \rangle + \frac{1}{2}\langle x - y, (A + \mu I)(x - y) \rangle - \frac{\mu}{2}\langle x - y, x - y \rangle. \end{aligned}$$

Since $\mu/2\langle x - y, x - y \rangle = \mu/2(\|x\|^2 + \|y\|^2) - \mu\langle x, y \rangle$, we get

$$-\frac{\mu}{2}\langle x - y, x - y \rangle - \mu\langle x - y, y \rangle = -\frac{\mu}{2}(\|x\|^2 + \|y\|^2) + \mu\|y\|^2 = -\frac{\mu}{2}(\|x\|^2 - \|y\|^2).$$

The proof is complete. $\square$

How to compute the initial point for restarting the DCA. a) If $\langle b, x^* \rangle > 0$, then taking $\bar{x} = -x^*$ we have $f(\bar{x}) < f(x^*)$. Indeed, in virtue of Lemma 4.2,

$$(22) \quad f(x) = f(x^*) - \frac{\lambda^*}{2}(\|x\|^2 - \|x^*\|^2) + \frac{1}{2}\langle x - x^*, (A + \lambda^* I)(x - x^*) \rangle.$$

This implies

$$f(\bar{x}) = f(x^*) + 2\langle (A + \lambda^* I)x^*, x^* \rangle = f(x^*) - 2\langle b, x^* \rangle < f(x^*).$$

So we can restart the DCA from the initial point $\bar{x}$.

b) If $\langle b, x^* \rangle \leq 0$, then we distinguish two cases:

(b.1). If either $\|x^*\| < r$ or $\|x^*\| = r$ and $u^T x^* \neq 0$, then we take $\bar{x} = x^* + \gamma u$, where $\gamma \neq 0$ is chosen such that $\|\bar{x}\| = r$. More precisely, γ is a nontrivial solution of the following equation:

$$\|u\|^2 \gamma^2 + 2u^T x^* \gamma + \|x^*\|^2 - r^2 = 0,$$

which is given explicitly by

$$(23) \quad \gamma = \begin{cases} -2(u^T x^*)/\|u\|^2 & \text{if } \|x^*\| = r, \\ \left(-(u^T x^*) \pm \sqrt{\Delta} \right) / \|u\|^2 & \text{if } \|x^*\| < r, \end{cases}$$

where $\Delta = (u^T x^*)^2 - \|u\|^2(\|x^*\|^2 - r^2)$.

So we can choose $\bar{x}$ as the initial point for restarting the DCA, since

$$f(\bar{x}) = f(x^*) + \frac{\gamma^2}{2}\langle u, (A + \lambda^* I)u \rangle < f(x^*).$$

Note that in (23) the greater γ^2 is, the smaller $f(\bar{x})$ becomes. Then the better value of γ is given by $\gamma = (-(u^T x^*) + \sqrt{\Delta})/\|u\|^2$ if $u^T x^* < 0$, $(-(u^T x^*) - \sqrt{\Delta})/\|u\|^2$ otherwise.

(b.2). If $\|x^*\| = r$, and $u^T x^* = 0$, then we find a new vector $v = u + \tau x^*$ with $\tau < 0$ such that $v^T (A + \lambda^* I)v < 0$ and $v^T x^* \neq 0$. Return to (b.1) with v instead of u .

To prove the existence of such a vector we will proceed as follows: we have $v^T x^* = u^T x^* + \tau \|x^*\|^2 = \tau \|x^*\|^2$. So $v^T x^* < 0$ if $\tau < 0$. Since

$$\begin{aligned} \langle v, (A + \lambda^* I)v \rangle &= \tau^2 \langle x^*, (A + \lambda^* I)x^* \rangle + 2\tau \langle u, (A + \lambda^* I)x^* \rangle + \langle u, (A + \lambda^* I)u \rangle \\ &= -\tau^2 \langle b, x^* \rangle - 2\tau \langle b, u \rangle + \langle u, (A + \lambda^* I)u \rangle \end{aligned}$$

we easily deduce the following properties:

- If $b^T x^* = 0$ and $b^T u \leq 0$, then $v^T (A + \lambda^* I)v < 0$ for all $\tau < 0$. If $b^T x^* = 0$ and $b^T u > 0$, then $v^T (A + \lambda^* I)v < 0$ for $0 > \tau > u^T (A + \lambda^* I)u / b^T u$.

- If $b^T x^* < 0$, then $v^T (A + \lambda^* I)v < 0$ for $\tau_1 < \tau < 0$, where τ_1 is the smallest root of the second degree polynomial of τ , $-b^T x^* \tau^2 - 2b^T u \tau + u^T (A + \lambda^* I)u$, i.e., $\tau_1 = [b^T u + ((b^T u)^2 + b^T x^* u^T (A + \lambda^* I)u)^{1/2}] / b^T x^*$.

Summing up, we conclude that if a Kuhn–Tucker point x^* produced by the DCA is not a global solution to (Q_1) then another point $\bar{x}$ (such that $\|\bar{x}\| \leq r$ and $f(\bar{x}) < f(x^*)$) is already available for restarting the DCA.

Remark. In case (b.2) the optimal choice for τ is a bit complicated because it deals with roots of a third degree polynomial of τ . Finally, we state here our global algorithm for solving (Q_1) in the nonconvex case.

Algorithm GDCA: (globally solving (Q_1)).

Initialization. Compute λ_1 and u by the IRLM.

Compute an approximation $\bar{\lambda}_n$ to λ_n by the IRLM.

Let $\rho := \bar{\lambda}_n + 0.1$, $\bar{x} \in C$, *stop* := *false*.

While *stop* = *false* **do**

1. Apply the DCA from starting point $x^o := \bar{x}$ to obtain x^* .

2. Let $\lambda^* = (-\langle x^*, Ax^* \rangle - \langle x^*, b \rangle) / r^2$.

If $\lambda^* \geq -\lambda_1$ **then** *stop* := *true*, x^* is a global solution of (Q_1)

else compute $\bar{x}$ such that $f(\bar{x}) < f(x^*)$ and return to 1.

end while

4.3. Relation between the DCA and the Goldstein–Levitin–Polyak gradient projection algorithm for solving (Q_1) in the convex case. In case the matrix A in (Q_1) is positive semidefinite, the Goldstein–Levitin–Polyak gradient projection algorithm applied to (Q_1) looks as follows:

$$(24) \quad x^{k+1} = P_E(x^k - \rho^{-1}(Ax^k + b)) = F(x^k).$$

The following well-known convergence result can be proved by using classical arguments of fixed point theory (B. Polyak [31]).

PROPOSITION 4.3. *Let A be positive semidefinite and $\rho > 0$. Then*

(i) $\|F(x) - F(x')\| \leq \max\{|1 - \rho^{-1}\lambda_n|, |1 - \rho^{-1}\lambda_1|\} \|x - x'\|$.

(ii) *If $\rho > \lambda_n/2$, then every limit point of $\{x^k\}$ is a solution of (Q_1) . Moreover, if the solution set of (Q_1) is finite, then the whole sequence $\{x^k\}$ converges to a solution of (Q_1) .*

(iii) *If A is positive definite and $0 < \rho^{-1} < 2\lambda_n^{-1}$, then F is a contraction with constant $q(\rho) < 1$. Moreover, $\bar{\rho} = (\lambda_n + \lambda_1)/2$ is the best choice which minimizes $q(\rho)$ over $]\lambda_n/2, +\infty[$:*

$$q(\bar{\rho}) = \frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}.$$

Remark. (i) The Goldstein–Levitin–Polyak gradient projection can be considered as a special case of the DCA applied to (Q_1) .

(ii) In the DCA ρ must be greater or equal to λ_n , while Proposition 4.3 requires only that $\rho > \lambda_n/2$.

(iii) The assertion (ii) of Proposition 4.3 is a direct consequence of Theorem 4.1.

(iv) Actually, the whole sequence $\{x^k\}$ converges to a solution of (Q_1) if this problem admits a solution. The proof of this result is based on the theory of maximal monotone operators.

4.4. Optimal choice of the regularization parameter ρ . In practice, the convergence rate of the DCA depends on the value ρ . Except for the case where the matrix A is positive definite for which the “theoretically optimal” value $(\lambda_1 + \lambda_n)/2$ (that is not “practically optimal,” however, as shown in Table 5) of ρ has been pointed out in Proposition 4.3, the general optimal choice of ρ is not easy. Numerical experiments proved that, in general, the closer to λ_n the positive parameter ρ is, the better the DCA converges. Such good parameters ρ belong to $\{\lambda > 0 : \lambda \in]\rho^*, \rho^* + \varepsilon]\}$, where $\varepsilon > 0$ is sufficiently small and $\rho^* = \lambda_n$ if $\lambda_n > 0$, 0 otherwise. In practice, such a ρ can be computed by using the quite inexpensive IRLM due to D.C. Sorensen [37]. We need only a low accuracy for an approximation $\tilde{\lambda}_n$ given by the IRLM of the largest eigenvalue λ_n of A . In other words, we are satisfied with an approximation $\tilde{\lambda}_n$ such that $\tilde{\lambda}_n \leq \lambda_n \leq \tilde{\lambda}_n + \Delta\tilde{\lambda}_n$, where the variation $\Delta\tilde{\lambda}_n$ is given by: if $|\tilde{\lambda}_n| = \alpha 10^k$ with $\alpha \in [0, 10[$ and k integer, then $\Delta\tilde{\lambda}_n = 2 \cdot 10^{k-1}$ when $\alpha \geq 5$, 10^{k-1} when $0 < \alpha < 5$, 0 when $\alpha = 0$.

Such an approximation $\tilde{\lambda}_n$ can be provided by the IRLM with the tolerance equal to 0.1 [37] at little cost. A good practical choice of ρ is then $\max\{\tilde{\lambda}_n + \Delta\tilde{\lambda}_n, 10^{-3}\}$.

5. Numerical experiments. In this section, we present some computational tests on the performance of the DCA for different sets of test problems. Our experiments are composed of five sets of tests. In the first set we compare the DCA with safeguarding [17] and dichotomy [7], [24] algorithms. The algorithms have been coded in PASCAL under Unix system and run on Sun SPARC-2 station with double precision. In the second set, we provide a comparison between the DCA and the approach proposed by S. A. Santos and D. C. Sorensen [35]. The third set reports the performance of the DCA and the gradient projection algorithm in the convex case. Finally, in the fourth and fifth sets we study the sensitivity of the DCA, respectively, for different choices of ρ and several sizes of the radius. For the last four sets the algorithms have been coded in FORTRAN 77 and run on Sun SPARC-10 station with double precision.

The stopping criterion of the DCA was actually $er \leq \epsilon$, where

$$(25) \quad er = \begin{cases} \|x^{k+1} - x^k\|^2 / \|x^k\|^2 & \text{if } \|x^k\| > 1, \\ \|x^{k+1} - x^k\|^2 & \text{otherwise,} \end{cases}$$

with $\epsilon = 10^{-6}$. The initial point of the DCA was always chosen as $x_i^0 = r/\sqrt{n}$, $i = 1, \dots, n$.

5.1. Comparison with safeguarding and dichotomy algorithms. In the first experiment we solved 150 problems (five problems per dimension n) whose data was randomly generated in a similar way as described in [17]. The matrix A is of the form $A = UDU^T$ for some orthogonal matrix U and a diagonal matrix D . Let $b = Ug$ for some vector g . The orthogonal matrix U was of the form $Q_1Q_2Q_3$ where $Q_j = I - 2w_jw_j^T/\|w_j\|^2$, $j = 1, 2, 3$ and the components w_j were random numbers in $(-1, 1)$. The elements of g were chosen as random numbers in $(-1, 1)$ while the diagonal elements of matrix D are random numbers in $(-5, 5)$. This choice

leads to the normal case. The hard case is generated by setting to zero all the components of g corresponding to the smallest element of D (that is λ_1) and then by taking r greater than $\|(A - \lambda_1 I)^+ b\|$ (here “ $+$ ” denotes the Moore–Penrose generalized inverse), i.e., $r > \|d\|$ where $d_i = g_i/(D_{ii} - \lambda_1)$ if $D_{ii} \neq \lambda_1$, 0 otherwise. A positive definite case (resp. positive semidefinite) was obtained by replacing $D_{ii} < 0$ by $|D_{ii}|$ (resp. 0). In the normal case, r is randomly chosen in $(1, 100)$. The values of ρ (in the DCA) were chosen as $0.25\|A\|_1$ or $0.35\|A\|_1$. For the safeguarding algorithm (Sfg) we took $\sigma_1 = 10^{-4}$, $\sigma_2 = 10^{-2}$. For the dichotomy algorithm (Dct) we chose $\varepsilon_3 = 0.001(\lambda_L^k + \lambda_U^k)$. The optimal values provided by the three algorithms are, to within 10^{-3} , equal.

Tables 1 and 2 indicate, respectively, the average iteration (IT) and the average CPU time in seconds (on Sun SPARC-2) of three algorithms in nonconvex and convex cases.

TABLE 1

Comparison with dichotomy and safeguarding algorithms, A is indefinite, the normal case (left) and the hard case (right).

n	DCA		Dct		Sfg	
	IT	time	IT	time	IT	time
100	53.2	1.3	6	2.5	7	3.2
200	113	11.3	6.2	28.4	7.4	32.6
250	102	13.5	5	42.1	7	60.6
300	75	15.5	6	85.6	7.4	95.3
350	73	20.4	5.2	122.2	7	161.6
400	59	23.6	5.2	172.0	6	189.7
450	77	80.6	4.4	250.2	6	270.2
500	67	90	6	497	6.4	502

n	DCA		Dct		Sfg	
	IT	time	IT	time	IT	time
100	83	2.2	8.6	4.7	10	5.5
200	222	22.2	9.2	30.3	11.2	37.9
250	180	37.5	9	62.1	11	75.4
300	178	45.0	7.2	75.0	12	130.4
350	190	74.0	6.4	129.2	12	244.9
400	279	122	5.6	181.4	11.4	355.1
450	187	139	6.6	316.6	11.2	539.6
500	234	144	7.6	475.7	11	684.9

TABLE 2

Comparison with dichotomy and safeguarding algorithms, A is positive definite (left) and A is positive semidefinite (right).

n	DCA		Dct		Sfg	
	IT	time	IT	time	IT	time
100	43	1.2	2.4	1.45	2.4	1.45
150	32	2.0	3	7.0	4.4	8.5
200	21	2.25	3.4	11.9	4	14.5
250	23	3.8	3.4	21.5	4	24.7
300	22	4.2	3	38.7	4	53.42
350	22	6.8	3	50.2	4	84.7
400	16	7.5	4.2	130	4	126.5

n	DCA		Dct		Sfg	
	IT	time	IT	time	IT	time
100	10	0.32	3	2.0	3	2.0
150	35	2.5	3	7.6	3	8.0
200	23	2.3	3.5	12.9	4	15.5
250	22	3.7	3	25.0	3	25.0
300	45	13.0	3	52.2	3.4	73.4
350	165	48.8	2	56.8	5	85.2
400	17	7.2	3	118	3	120

5.2. Comparison with the new matrix-free algorithm. In the second experiment we solved two different families of 112 problems by the DCA and the new matrix-free algorithm (NMFA). We used the code in MATLAB of S. A. Santos and D. C. Sorensen [35] to generate the data. First, the matrix A takes the form $L - 5I$, where L is the standard 2-D discrete Laplacian on the unit square based upon a 5-point stencil with equally-spaced meshpoints. The shift of $-5I$ was introduced to make A indefinite. The vector b was randomly generated with entries uniformly distributed on $(0, 1)$. The hard case is generated by performing the operation $b \leftarrow b - q_1(q_1^T b)$ to orthogonalize the vector b against q_1 , the eigenvector of A corresponding to the smallest eigenvalue λ_1 . The radius was always equal to 100.

In the second family of problems, the matrix A is of the form UDU^T and generated as in subsection 5.6. of [35]. D was a diagonal matrix and $U = I - 2uu^T$, $u^T u = 1$. The elements of $D = \text{diag}(d_1, \dots, d_n)$ were randomly generated and uniformly distributed

on $(-5, 5)$. The entries of u and b were randomly selected from a uniform distribution on $(-0.5, 0.5)$. The vector u was normalized to have unit length. As in the first family, an orthogonalization of b against q_1 generated a hard case. A noise vector of norm 10^{-8} was added to b . To ensure the hard case the radius was chosen as $r = 2r_{min}$ where $r_{min} = \|(A - d_1 I)^+ b\|$. In the normal case the radius was fixed at $r = 100$ when $n = 256, n = 576, n = 1024$ and $r = 20$ when $n = 100$.

We used the IRLM implemented in the package ARPACK [12], [37] for computing the two smallest eigenpairs at each iteration in the NMFA. For the IRLM we choose nine Lanczos basis vectors, six shifts on each implicit restart, and different tolerances at each case of the two families. In the first family ($A = L - 5I$) the initial tolerance was $\varepsilon_R = 0.5$ for all dimensions in both cases, except $\varepsilon_R = 0.1$ when $n = 1024$ in the normal case. Subsequently, in the normal case $\varepsilon_R = \max\{\min\{\varepsilon_R, |r - \|x^k\|/r|\}, 0.075\}$ when $n = 100, n = 256, n = 576$; $\varepsilon_R = \max\{\min\{\varepsilon_R, |r - \|x^k\|/r|\}, 0.01\}$ when $n = 1024$ and in the hard case; $\varepsilon_R = \max\{\min\{\varepsilon_R, |r - \|x^k\|/1000r|, |\tau^2(z^T A z - \lambda_k)/(1000(b^T x^k + \lambda_k r^2))|, 0.01\}\}$ when $n = 100, n = 256, n = 576$, $\varepsilon_R = \max\{\min\{\varepsilon_R, |r - \|x^k\|/1000r|, |\tau^2(z^T A z - \lambda_k)/(1000(b^T x^k + \lambda_k r^2))|, 0.001\}\}$ when $n = 1024$. In the second family ($A = UDU^T$) the initial tolerance was $\varepsilon_R = 10^{-4}$ (resp., $\varepsilon_R = 10^{-2}$) when $n = 100$, $\varepsilon_R = 10^{-6}$ (resp., $\varepsilon_R = 10^{-3}$) when $n = 256, n = 576$, and $\varepsilon_R = 10^{-7}$ (resp., $\varepsilon_R = 10^{-4}$) when $n = 1024$ in the hard case (resp., the normal case). Subsequently, $\varepsilon_R = \min\{\varepsilon_R, |r - \|x^k\|/1000r|, |\tau^2(z^T A z - \lambda_k)/(1000(b^T x^k + \lambda_k r^2))|\}$ (resp., $\varepsilon_R = \min\{\varepsilon_R, \sqrt{|r - \|x^k\|/r|}, |\tau^2(z^T A z - \lambda_k)/(b^T x^k + \lambda_k r^2)|\}$). For a fixed tolerance ε_R , the stopping condition in the IRLM is $(\|f\| \|e_j^T y\|)^2 \leq \varepsilon_R |\mu| G(j)$ (cf. [19, pp. 206, 222]). In our experiments $j = 3$. The IRLM was started with $v = (1, \dots, 1)^T / \sqrt{n+1}$ and then the previously calculated eigenvector corresponding to the smallest eigenvalue was used as the initial vector. The parameters for the NMFA were the following: $\varepsilon_\delta = \varepsilon_{hc} = 10^{-6}$, $\varepsilon_\nu = 10^{-2}$, $\alpha_0 = \min\{0, \alpha_U\}$ and the initial of δ_S was $\delta_S = \min\{A_{ii} : i = 1, \dots, n\} + 0.5$ in the hard case with $A = UDU^T$ and $\delta_S = \min\{A_{ii} : i = 1, \dots, n\}$ in other cases. (See [35] for these notations.)

For computing ρ in the DCA we also applied the IRLM to the matrix A with the tolerance $\varepsilon_R = 0.5$ in the first family ($A = L - 5I$) and $\varepsilon_R = 0.1$ in the second family ($A = UDU^T$). So we obtained an approximation $\tilde{\lambda}_n$ of the largest eigenvalue λ_n of A and took $\rho = \tilde{\lambda}_n + 0.1$ (resp. $\rho = \tilde{\lambda}_n + 0.5$) when $A = L - 5I$ (resp. $A = UDU^T$).

For each case (normal and hard cases) seven problems per dimension were generated with different seeds for each family and solved by both Santos and Sorensen's algorithm (NMFA) and our code (DCA). The average results are indicated in Tables 3 and 4. In these tables we used the following notation: IT stands for the number of iterations; MVP stands for the number of matrix-vector products; KTC = $\|b + (A + \lambda^* I)x^*\| / \|b\|$ (this presents the Kuhn-Tucker condition); time: CPU time in seconds (on Sun SPARC-10). The value λ^* is the *average* of the obtained multipliers provided by each algorithm. Note that for the DCA the number of required matrix-vector products is exactly the sum of the number of iterations in the DCA and the number of matrix-vector products in the IRLM. In our experiments we observe from numerical results that the last quantity was always equal to nine. The quantity KTC in the DCA was equal to zero because λ^* was deduced from the expression $(A + \lambda^* I)x^* = -b$. CPU time of the DCA was presented in two parts: the first (resp. second) part indicated CPU time of the IRLM (resp. DCA).

5.3. Comparison with the gradient projection algorithm in the convex case. In the third experiment we solved 56 problems whose data was generated as

TABLE 3
Comparison with the new matrix-free method, $A = L - 5I$.

n	Type	Algorithm	IT	MVP	$\frac{r - \ x^*\ }{r}$	KTC	λ^*	time
100	normal case	NMFA	7.4	89.4	-3.10^{-8}	7.10^{-5}	4.8822	0.3
		DCA	47.6	56.6	-3.10^{-17}	0.0	4.7939	0.03+0.02
100	hard case	NMFA	12.6	298.2	-9.10^{-17}	3.10^{-3}	4.8340	0.9
		DCA	45.2	54.2	6.10^{-17}	0.0	4.8381	0.03+0.02
256	normal case	NMFA	7.8	131.4	-9.10^{-8}	2.10^{-6}	5.0013	0.7
		DCA	108.6	117.6	-2.10^{-16}	0.0	4.8631	0.04+ 0.1
256	hard case	NMFA	9.4	383.4	2.10^{-17}	9.10^{-4}	4.9319	1.7
		DCA	86.8	95.6	5.10^{-33}	0.0	4.9325	0.04+0.1
576	normal case	NMFA	8.2	135	10^{-7}	8.10^{-6}	5.0709	1.5
		DCA	471.8	480.8	-2.10^{-16}	0.0	5.0701	0.07+1.3
576	hard case	NMFA	10.8	678	2.10^{-16}	10^{-2}	4.9684	5.2
		DCA	166.6	175.6	-4.10^{-16}	0.0	4.9711	0.07 + 0.5
1024	normal case	NMFA	7.4	187.8	4.10^{-8}	4.10^{-7}	5.1212	4.9
		DCA	346.8	355.8	2.10^{-16}	0.0	5.1188	0.1+1.8
1024	hard case	NMFA	11.6	1490	-2.10^{-7}	2.10^{-2}	4.9818	20.1
		DCA	315.2	324.2	-2.10^{-16}	0.0	4.9907	0.1+1.6

TABLE 4
Comparison with the new matrix-free method, $A = UDU^T$.

n	Type	Algorithm	IT	MVP	$\frac{r - \ x^*\ }{r}$	KTC	λ^*	time
100	normal case	NMFA	6.6	313.8	4.10^{-8}	6.10^{-8}	4.970	3.2
		DCA	249	258	-7.10^{-17}	0.0	4.959	0.1+2.2
100	hard case	NMFA	8.8	909.6	-3.10^{-17}	10^{-2}	4.9492	9.1
		DCA	307.6	316.6	-2.10^{-16}	0.0	4.9469	0.1+2.7
256	normal case	NMFA	8	962.4	-7.10^{-9}	5.10^{-4}	4.9748	48.3
		DCA	316.6	325.6	5.10^{-16}	0.0	4.9698	0.5+18.2
256	hard case	NMFA	6.6	1447.8	2.10^{-16}	8.10^{-3}	4.9730	144.3
		DCA	316.8	325.8	7.10^{-17}	0.0	4.9717	0.5+18.4
576	normal case	NMFA	9.3	1658.3	-2.10^{-7}	10^{-8}	4.9852	391.9
		DCA	348	357	-4.10^{-16}	0.0	4.9812	3.2+124.9
576	hard case	NMFA	8.4	2242.8	-6.10^{-17}	4.10^{-3}	4.9843	1347.3
		DCA	343.4	352.3	-8.10^{-16}	0.0	4.9809	3.3+123.9
1024	normal case	NMFA	9.8	2318.3	2.10^{-7}	4.10^{-9}	4.9876	2209.8
		DCA	431	440	-6.10^{-16}	0.0	4.9806	13.2+476.4
1024	hard case	NMFA	10.3	3279	-8.10^{-16}	10^{-2}	4.9919	4381.6
		DCA	439.6	448.6	9.10^{-16}	0.0	4.9861	13.2+576.4

in subsection 5.1. We considered two types of matrices for A : positive semidefinite (p.s.d.) and positive definite (p.d.) with $n = 250$ and $n = 500$. The diagonal elements of D were in $(-5, 5)$ when $n = 250$ and in $(-50, 50)$ when $n = 500$. The elements of g were random numbers in $(-1, 1)$. As in subsection 5.1, a p.d. (resp., p.s.d.) matrix $A = UDU^T$ was obtained by replacing $D_{ii} < 0$ by $|D_{ii}|$ (resp., 0). We considered two cases (normal and hard case) when A is a p.s.d. matrix. For each type of A seven problems per dimension and per case were solved by the gradient projection algorithm (GPA) and by the DCA. In the GPA we took $\rho = 1/2\lambda_n + 0.1$ if A is p.s.d. and the theoretically optimal $\rho = 1/2(\lambda_1 + \lambda_n)$ if A is p.d. In the DCA we took $\rho = \lambda_n$. The average result is reported in Table 5.

TABLE 5
Comparison with the gradient projection method in the convex case.

n	Type of A	Algorithm	IT	$\frac{r-\ x^*\ }{r}$	λ^*	time
250	A p.d	GPA	640.4	-2.10^{-16}	0.5539	50.8
		DCA	250.2	0.06	0.5533	20.0
250	A p.s.d normal case	GPA	48	-10^{-16}	1.0230	4.1
		DCA	79	2.10^{-16}	1.0230	6.4
250	A p.s.d hard case	GPA	1360.4	0.01	0.0009	108.0
		DCA	250.8	0.02	0.0011	20.3
500	A p.d	GPA	2573.2	0.2841	0.0008	877.2
		DCA	294.6	0.3866	0.00172	101.4
500	A p.s.d normal case	GPA	225.6	-3.10^{-16}	0.4827	80.4
		DCA	305	-2.10^{-16}	0.4800	108.0
500	A p.s.d hard case	GPA	742.6	0.05	0.0074	254.4
		DCA	349.4	0.1	0.0095	120.1

5.4. Different choices for ρ . In the fourth experiment we solved 20 problems (ten per dimension) of the second family generated exactly as in subsection 5.2. We considered the normal case when $n = 256$ and the hard case when $n = 1024$. In the first case the radius was equal to 100 and in the last case it varied in the interval (110, 701). Each of these problems was solved five times, with $\rho = \|A\|_1$, $\rho = 1/2\|A\|_1$, $\rho = 1/3\|A\|_1$, $\rho = 5$, and $\rho = \tilde{\lambda}_n + 0.5$. The value $\tilde{\lambda}_n$ was an approximation to the largest eigenvalue of A calculated by the IRLM with the tolerance $tol = 0.1$. In these problems, $\|A\|_1$ were in the interval (18, 19.5). The average result is indicated in Table 6. CPU time (on Sun SPARC-10) in case $\rho = \tilde{\lambda}_n + 0.5$ was presented in two parts as in Tables 3 and 4. The quantity avOPT presented the average of optimal values.

TABLE 6
Average behavior for different choices of ρ .

n	ρ	$\frac{r-\ x^*\ }{r}$	avOPT	λ^*	IT	time
256 normal case	$\ A\ _1$	4.10^{-16}	-24823.7	4.9626	411	23.3
	$0.5\ A\ _1$	-3.10^{-16}	-24851.1	4.9682	362.4	20.6
	$0.3\ A\ _1$	3.10^{-16}	-24858.5	4.9697	319.4	18.4
	5.0	-1.10^{-16}	-24859.4	4.9699	311.6	18.4
	$\tilde{\lambda}_n + 0.5$	5.10^{-16}	-24858.9	4.96988	316.6	0.5+18.2
1024 hard case	$\ A\ _1$	-2.10^{-16}	-471484.3	4.9691	503	551.4
	$0.5\ A\ _1$	8.10^{-15}	-496199.1	4.9751	455.2	515.2
	$0.3\ A\ _1$	-10^{-15}	-496438.5	4.9795	438	487.3
	5.0	3.10^{-16}	-496481.4	4.9797	433.7	472.7
	$\tilde{\lambda}_n + 0.5$	3.10^{-16}	-496481.4	4.9797	433.7	13.2 + 502.7

5.5. Different sizes for the radius. In the last experiment (Table 7) we solved 10 problems of the second family mentioned above. We considered here the normal case with $n = 500$. The radius varied by a factor of 10 through the values 100, ..., 0.001. ρ was always equal to 5.

5.6. Comments on computational results. (i) For every test problem the DCA (without restarting procedure) always provided a global solution.

TABLE 7
Average behavior for different sizes of the radius.

$r \rightarrow$	100	10	1	0.1	0.01	0.001
IT	354	314	18	5	3	2.2
time	120.6	107.1	6.7	2.5	1.8	1.5

(ii) The parameter ρ as well as the d.c. representation of the objective function have an important influence on the convergence rate of the DCA. The practical choice (subsection 4.6) of ρ seems to be well suited (Table 6). We observe (see Table 5) the following:

- If A is positive definite, the choice $\rho = \lambda_n$ is much better than the theoretically optimal choice $\rho = (\lambda_1 + \lambda_n)/2$ given in Proposition 1. The ratio of execution times is 2.5 for $n = 250$ and 8.7 for $n = 500$.

- If A is positive semidefinite, the choice $\rho = \lambda_n$ is better than the choice $\rho = 0.5\lambda_n + 0.1$ in the hard case. The ratio of execution times is 5.4 when $n = 250$ and 2 when $n = 500$. The situation reverses in the normal case (1.6 when $n = 250$ and 1.35 when $n = 500$).

(iii) The DCA is faster than the Sfg. Table 1 indicates that in the normal case (resp., hard case) with A being indefinite the ratio of execution times varies between 2.4 and 8 (resp., 1.7 and 4.7). In the convex case Table 2 shows that the ratio of execution times varies between 1.2 and 16.8 (resp., 1.7 and 16.6) when A is positive definite (resp., positive semidefinite).

(iv) The DCA is faster than the NMFA. In the case where $A = L - 5I$, Table 3 indicates that the ratio of execution times belongs to the interval $]1, 18[$. It is more significant in the hard case (equal to 18 when $n = 100$, 12 when $n = 256$, 10 when $n = 576$ and 11.8 when $n = 1024$). In the normal case the ratio is smaller (equal to 6 when $n = 100$, 5 when $n = 256$ and 2.6 when $n = 1024$).

In the case where $A = UDU^T$, Table 4 shows similar results. In the normal case the ratio of execution times is 1.4 when $n = 100$, 2.6 when $n = 256$, 3 when $n = 576$, and 4.5 when $n = 1024$. It becomes (in the hard case) 3.2 when $n = 100$, 7.6 when $n = 256$, 10.6 when $n = 576$, and 7.4 when $n = 1024$.

(v) One of the referees suggested the example: $A = \text{Diag}(1, -1)$, $b = (1, 1)$, $r = 2$, and $\rho = 1.1$, where the pure DCA (i.e., without the restarting procedure) with the choice $x^o = (r/\sqrt{n}, r/\sqrt{n}) = (\sqrt{2}, \sqrt{2})$ does not lead to a global solution. However, we observe that the random choice of $x^o : x_i^o = 2[\text{random}(100) - 0.5]$, $i = 1, 2$ yields a global one.

6. Conclusion. We have presented a thorough study of the d.c. optimization, its algorithm DCA, and the application of the DCA to solving the trust-region subproblems.

By its very approach the DCA is completely different from other available methods for the trust-region problem [17], [24], [32], [35], [38]. The DCA is quite simple and requires matrix-vector products only. Moreover, it indifferently treats the normal and hard cases of the trust-region subproblem. Theoretically, it has been stated that the DCA converges to a Kuhn-Tucker point of the trust-region subproblem. We observe, however, that in all numerical examples we tried, convergence occurred to a global minimizer. The foundations of the d.c. optimization theory enable us to conjecture that the DCA converges to a local minimizer (of the trust-region subproblem) and so, most times, to a global one according to Martinez's results. In the absence of a

theoretical proof of the conjecture, we have pointed out a simple numerical procedure to restart the DCA with a new initial point-in-case when the solution provided by the DCA is not global. This restarting procedure needs computing the smallest eigenvalue of the matrix A (in the trust-region subproblem) and a corresponding eigenvector and uses for that the quite inexpensive *Implicitly Restarted Lanczos Method* introduced by D. C. Sorensen [37]. Since the IRLM meets the requirements of limited storage and reliance only on matrix–vector products, the DCA with at most $2m + 2$ restartings (Theorem 4) requires only matrix–vector products too and converges to a global solution. In practice the DCA rarely has recourse to the restarting procedure. From the computational viewpoint, a lot of our numerical experiments proved the robustness and efficiency of the DCA with respect to other well-known algorithms, especially in the large scale trust-region subproblem.

Appendix. Proof of the basic convergence theorem of the DCA for general d.c. programming. First, we need the following results.

PROPOSITION A.1. *Suppose that the sequences $\{x^k\}$ and $\{y^k\}$ are generated by the simplified DCA. Then we have*

$$(i) \quad (g - h)(x^{k+1}) \leq (h^* - g^*)(y^k) - \frac{\rho_2}{2} \|dx^k\|^2 \leq (g - h)(x^k) - \frac{\rho_1 + \rho_2}{2} \|dx^k\|^2.$$

The equality $(g - h)(x^{k+1}) = (g - h)(x^k)$ holds if and only if

$$x^k \in \partial g^*(y^k), y^k \in \partial h(x^{k+1}) \text{ and } (\rho_1 + \rho_2) \|dx^k\| = 0.$$

(ii) *Similarly, by duality we have*

$$(h^* - g^*)(y^{k+1}) \leq (g - h)(x^{k+1}) - \frac{\rho_1^*}{2} \|dy^k\|^2 \leq (h^* - g^*)(y^k) - \frac{\rho_1^* + \rho_2^*}{2} \|dy^k\|^2.$$

The equality $(h^* - g^*)(y^{k+1}) = (h^* - g^*)(y^k)$ holds if and only if

$$x^{k+1} \in \partial g^*(y^{k+1}), y^k \in \partial h(x^{k+1}) \text{ and } (\rho_1^* + \rho_2^*) \|dy^k\| = 0.$$

Proof of Proposition A.1. (i) Since $y^k \in \partial h(x^k)$ we have $h(x^{k+1}) \geq h(x^k) + \langle x^{k+1} - x^k, y^k \rangle + \rho_2/2 \|dx^k\|^2$. Hence,

$$(26) \quad (g - h)(x^{k+1}) \leq g(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - h(x^k) - \frac{\rho_2}{2} \|dx^k\|^2.$$

Likewise, $x^{k+1} \in \partial g^*(y^k)$ implies

$$g(x^k) \geq g(x^{k+1}) + \langle x^k - x^{k+1}, y^k \rangle + \frac{\rho_1}{2} \|dx^k\|^2.$$

So,

$$(27) \quad g(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - h(x^k) \leq (g - h)(x^k) - \frac{\rho_1}{2} \|dx^k\|^2.$$

On the other hand,

$$(28) \quad x^{k+1} \in \partial g^*(y^k) \Leftrightarrow \langle x^{k+1}, y^k \rangle = g(x^{k+1}) + g^*(y^k),$$

$$(29) \quad y^k \in \partial h(x^k) \Leftrightarrow \langle x^k, y^k \rangle = h(x^k) + h^*(y^k).$$

Thus,

$$(30) \quad g(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - h(x^k) = h^*(y^k) - g^*(y^k).$$

Finally, combining (26), (27), and (30), we get

$$(31) \quad (g - h)(x^{k+1}) \leq (h^* - g^*)(y^k) - \frac{\rho_2}{2} \|dx^k\|^2 \leq (g - h)(x^k) - \frac{\rho_1 + \rho_2}{2} \|dx^k\|^2.$$

If $\rho_1 + \rho_2 > 0$, the last statement of (i) is an immediate consequence of the equality $dx^k = 0$ and the construction of the sequences $\{x^k\}$ and $\{y^k\}$.

It is clear that there exist ρ_1 and ρ_2 such that $\rho_1 + \rho_2 > 0$ if and only if $\rho(h) + \rho(g) > 0$. In case $\rho(h) = \rho(g) = 0$, (29) implies the equivalence between $(g - h)(x^{k+1}) = (g - h)(x^k)$ and the combination of (32) and (33)

$$(32) \quad (g - h)(x^{k+1}) = (h^* - g^*)(y^k),$$

$$(33) \quad (h^* - g^*)(y^k) = (g - h)(x^k).$$

We then deduce from (28) and (32) that $h(x^{k+1}) + h^*(y^k) = \langle x^{k+1}, y^k \rangle$, i.e., $y^k \in \partial h(x^{k+1})$. Similarly, (29) and (33) give $g(x^k) + g^*(y^k) = \langle x^k, y^k \rangle$, i.e., $x^k \in \partial g^*(y^k)$. Property (ii) is proved analogously. $\square$

The following result is an important consequence of Proposition A.1.

COROLLARY A.2.

$$(i) \quad (g - h)(x^{k+1}) \leq (h^* - g^*)(y^k) - \frac{\rho_2}{2} \|dx^k\|^2 \leq (g - h)(x^k) - \left[\frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2}{2} \|dx^k\|^2 \right].$$

$$(ii) \quad (g - h)(x^{k+1}) \leq (h^* - g^*)(y^k) - \frac{\rho_2^*}{2} \|dy^k\|^2 \leq (g - h)(x^k) - \left[\frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2^*}{2} \|dy^k\|^2 \right].$$

The equality $(g - h)(x^{k+1}) = (g - h)(x^k)$ holds if and only if

$$x^k \in \partial g^*(y^k), \quad y^k \in \partial h(x^{k+1}) \quad \text{and} \quad (\rho_1 + \rho_2) dx^k = \rho_1^* dy^{k-1} = \rho_2^* dy^k = 0.$$

Similarly, by duality we have

$$(iii) \quad (h^* - g^*)(y^{k+1}) \leq (g - h)(x^{k+1}) - \frac{\rho_1^*}{2} \|dy^k\|^2 \leq (h^* - g^*)(y^k) - \left[\frac{\rho_1^*}{2} \|dy^k\|^2 + \frac{\rho_2}{2} \|dx^k\|^2 \right].$$

$$(iv) \quad (h^* - g^*)(y^{k+1}) \leq (g - h)(x^{k+1}) - \frac{\rho_1}{2} \|dx^{k+1}\|^2 \leq (h^* - g^*)(y^k) - \left[\frac{\rho_1}{2} \|dx^{k+1}\|^2 + \frac{\rho_2}{2} \|dx^k\|^2 \right].$$

The equality $(h^* - g^*)(y^{k+1}) = (h^* - g^*)(y^k)$ holds if and only if

$$x^{k+1} \in \partial g^*(y^{k+1}), \quad y^k \in \partial h(x^{k+1}) \quad \text{and} \quad (\rho_1^* + \rho_2^*) dy^k = \rho_2 dx^k = \rho_1 dx^{k+1} = 0.$$

Proof. The inequalities in (i) and (ii) are easily deduced from Properties (i) and (ii) of Proposition A.1. The inequalities in (iii) and (iv) can be shown by the same arguments as in the proof of Proposition A.1. $\square$

We are now in a position to demonstrate Theorem 3.7.

Proof of Theorem 3.7. Properties (i) and (ii) are proved analogously; therefore, we give here the proof for (i) only.

The first inequality of (i) is an immediate consequence of (i) and (ii) of Corollary A.2.

- If $\rho_2 \|dx^k\|^2 \leq \rho_2^* \|dy^k\|^2$, then it follows from (ii) of Corollary A.2 that

$$\begin{aligned} (h^* - g^*)(y^k) - \max \left\{ \frac{\rho_2}{2} \|dx^k\|^2, \frac{\rho_2^*}{2} \|dy^k\|^2 \right\} &= (h^* - g^*)(y^k) - \frac{\rho_2^*}{2} \|dy^k\|^2 \\ &\leq (g - h)(x^k) - \left\{ \frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2^*}{2} \|dy^k\|^2 \right\}. \end{aligned}$$

Similarly, Property (i) of Corollary A.2 implies

$$\begin{aligned} (h^* - g^*)(y^k) - \frac{\rho_2^*}{2} \|dy^k\|^2 &\leq (h^* - g^*)(y^k) - \frac{\rho_2}{2} \|dx^k\|^2 \\ &\leq (g - h)(x^k) - \left[\frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2}{2} \|dx^k\|^2 \right]. \end{aligned}$$

On the other hand, by (i) of Proposition A.1,

$$(h^* - g^*)(y^k) - \frac{\rho_2^*}{2} \|dy^k\|^2 \leq (h^* - g^*)(y^k) - \frac{\rho_2}{2} \|dx^k\|^2 \leq (g - h)(x^k) - \frac{\rho_1 + \rho_2}{2} \|dx^k\|^2.$$

Combining these inequalities, we get the second inequality of (i).

- If $\rho_2^* \|dy^k\|^2 \leq \rho_2 \|dx^k\|^2$, then by using the same arguments we can easily show the second inequality of (i). The first property of (iii) is evident. We will prove the last one.

Taking (30) and (i) into account, we have

$$\lim_{k \rightarrow +\infty} (g - h)(x^{k+1}) = \lim_{k \rightarrow +\infty} \{g(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - h(x^k)\} = \lim_{k \rightarrow +\infty} (g - h)(x^k).$$

The second equality implies $\lim_{k \rightarrow +\infty} \{g(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - g(x^k)\} = 0$, i.e., $\lim_{k \rightarrow +\infty} \{g(x^k) + g^*(y^k) - \langle x^k, y^k \rangle\} = 0$, since $x^{k+1} \in \partial g^*(y^k)$.

Likewise, it results from the first equality that $\lim_{k \rightarrow +\infty} \{h(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - h(x^k)\} = 0$, i.e., $\lim_{k \rightarrow +\infty} \{h(x^{k+1}) + h^*(y^k) - \langle x^{k+1}, y^k \rangle\} = 0$ since $y^k \in \partial h(x^k)$.

(iv) We assume α is finite and the sequences $\{x^k\}$ and $\{y^k\}$ are bounded. Let x^* be a limit point of $\{x^k\}$. For the sake of simplicity we write $\lim_{k \rightarrow +\infty} x^k = x^*$. We can suppose (by extracting a subsequence if necessary) that the sequence $\{y^k\}$ converges to $y^* \in \partial h(x^*)$. Property (iii) then implies

$$\lim_{k \rightarrow +\infty} \{g(x^k) + g^*(y^k)\} = \lim_{k \rightarrow +\infty} \langle x^k, y^k \rangle = \langle x^*, y^* \rangle.$$

Let $\theta(x, y) = g(x) + g^*(y)$ for $(x, y) \in X \times Y$. It is clear that $\theta \in \Gamma_o(X \times Y)$. Then the lower semicontinuity of θ implies

$$\theta(x^*, y^*) \leq \lim_{k \rightarrow +\infty} \inf \theta(x^k, y^k) = \lim_{k \rightarrow +\infty} \theta(x^k, y^k) = \langle x^*, y^* \rangle,$$

i.e., $\theta(x^*, y^*) = g(x^*) + g^*(y^*) = \langle x^*, y^* \rangle$.

In other words, $y^* \in \partial g(x^*)$. According to Lemma A.3 stated below we have

$$\lim_{k \rightarrow +\infty} h(x^k) = h(x^*) \text{ and } \lim_{k \rightarrow +\infty} h^*(y^k) = h^*(y^*) \text{ since } y^k \in \partial h(x^k), x^k \rightarrow x^* \text{ and } y^k \rightarrow y^*.$$

Hence, in virtue of (iii),

$$\lim_{k \rightarrow +\infty} (g - h)(x^k) = \lim_{k \rightarrow +\infty} g(x^k) - \lim_{k \rightarrow +\infty} h(x^k) = \lim_{k \rightarrow +\infty} g(x^k) - h(x^*) = \beta,$$

$$\lim_{k \rightarrow +\infty} (h^* - g^*)(y^k) = \lim_{k \rightarrow +\infty} h^*(y^k) - \lim_{k \rightarrow +\infty} g^*(y^k) = h^*(y^*) - \lim_{k \rightarrow +\infty} g^*(y^k) = \beta.$$

It then suffices to show that

$$\lim_{k \rightarrow +\infty} g(x^k) = g(x^*); \quad \lim_{k \rightarrow +\infty} g^*(y^k) = g^*(y^*).$$

Since $\lim_{k \rightarrow +\infty} g(x^k)$ and $\lim_{k \rightarrow +\infty} g^*(y^k)$ exist, (iii) implies

$$g(x^*) + g^*(y^*) = \lim_{k \rightarrow +\infty} \{g(x^k) + g^*(y^k)\} = \lim_{k \rightarrow +\infty} g(x^k) + \lim_{k \rightarrow +\infty} g^*(y^k).$$

Further, because of the lower semicontinuity of g and g^* ,

$$\lim_{k \rightarrow +\infty} g(x^k) = \lim_{k \rightarrow +\infty} \inf g(x^k) \geq g(x^*), \quad \lim_{k \rightarrow +\infty} g^*(y^k) = \lim_{k \rightarrow +\infty} \inf g^*(y^k) \geq g^*(y^*).$$

The former equalities imply that these last inequalities are in fact equalities. The proof of Theorem 3.7 is complete. $\square$

LEMMA A.3. *Let $h \in \Gamma_o(X)$ and $\{x^k\}$ be a sequence of elements in X such that*

- (i) $x^k \rightarrow x^*$;
- (ii) *there exists a bounded sequence $\{y^k\}$ with $y^k \in \partial h(x^k)$;*
- (iii) $\partial h(x^*)$ *is nonempty.*

Then $\lim_{k \rightarrow +\infty} h(x^k) = h(x^*)$.

Proof. Indeed, let $y^* \in \partial h(x^*)$, then $h(x^k) \geq h(x^*) + \langle x^k - x^*, y^* \rangle \quad \forall k$. Since $y^k \in \partial h(x^k)$ we have $h(x^*) \geq h(x^k) + \langle x^* - x^k, y^k \rangle \quad \forall k$. Hence, $h(x^k) \leq h(x^*) + \langle x^k - x^*, y^k \rangle \quad \forall k$. As $x^k \rightarrow x^*$, we have $\lim_{k \rightarrow +\infty} \langle x^k - x^*, y^* \rangle = 0$. Moreover, $\lim_{k \rightarrow +\infty} \langle x^k - x^*, y^k \rangle = 0$, since the sequence $\{y^k\}$ is bounded. Consequently, $\lim_{k \rightarrow +\infty} h(x^k) = h(x^*)$. $\square$

Acknowledgments. The authors are grateful to the referees and to Professor J. M. Martinez for their helpful comments and suggestions as well as for their example in 5.6 (v) which have improved the presentation of the revised paper. Thanks also go to Dr. S. A. Santos and Professor D. C. Sorensen for making their MATLAB code available to us for experimentation.

REFERENCES

- [1] J. R. CLERMONT, M. E. DE LA LANDE, AND T. PHAM DINH, *Analysis of plane and axisymmetric flows of incompressible fluids with the stream tube method: Numerical simulation by trust region algorithm*, Internat. J. Numer. Methods Fluids, 13 (1991), pp. 371–339.
- [2] R. FLETCHER, *Practical Methods of Optimization*, 2nd ed., Wiley-Interscience, New York, 1991.
- [3] G. E. FORSYTHE AND G. H. GOLUB, *On the stationary values of a second-degree polynomial on the unit sphere*, J. Soc. Indust. Appl. Math., 13 (1965), pp. 1050–1068.
- [4] W. GANDER, G. H. GOLUB, AND U. VON MATT, *A constrained eigenvalue problem*, Linear Algebra Appl., 114/115 (1989), pp. 815–839.
- [5] D. M. GAY, *Computing optimal locally constrained steps*, SIAM J. Sci. Statist. Comput., 2 (1981), pp. 186–197.
- [6] G. H. GOLUB AND U. VON MATT, *Quadratically constrained least squares and quadratic problems*, Numer. Math., 59 (1991), pp. 561–580.

- [7] L. T. HOAI AN, *Analyse numérique des algorithmes de l'Optimisation d.c. Approches locales et globales. Code et simulations numériques en grande dimension. Applications*, Thèse de Doctorat de l'Université de Rouen, France, 1994.
- [8] L. T. HOAI AN AND T. PHAM DINH, *Solving a class of linearly constrained indefinite quadratic problems by D.c. algorithms*, J. Global Optim., 11 (1997), pp. 253–285.
- [9] J. B. HIRIART-URRUTY, *From convex optimization to non convex optimization. Part I: Necessary and sufficient conditions for global optimality*, in Nonsmooth Optimization and Related Topics, Ettore Majorana International Sci. Ser. Phys. Sci. 43, Plenum Press, New York, 1988.
- [10] R. HORST AND H. TUY, *Global Optimization (Deterministic Approaches)*, Springer-Verlag, Berlin, 1993.
- [11] P. J. LAURENT, *Approximation et Optimisation*, Hermann, Paris, 1972.
- [12] R. LEHOUCQ, D. C. SORESENSEN, AND P. A. VU, *ARPACK: An implementation of the implicitly restarted Arnoldi iteration that computes some of the eigenvalues and eigenvectors of a large sparse matrix*, available online from netlib@ornl.gov under the directory scalapack.
- [13] S. LUCIDI, L. PALAGI, AND M. ROMA, *On some properties of quadratic programs with a convex quadratic constraint*, SIAM J. Optim., 8 (1998), pp. 105–122.
- [14] P. MAHEY AND T. PHAM DINH, *Proximal decomposition of the graph of maximal monotone operator*, SIAM J. Optim., 5 (1995), pp. 454–468.
- [15] J. M. MARTINEZ, *Local minimizers of quadratic functions on Euclidean balls and spheres*, SIAM J. Optim., 4 (1994), pp. 159–176.
- [16] J. J. MORE, *Recent developments in algorithm and software for trust region methods*, in Mathematical Programming, The State of the Art, Springer-Verlag, Berlin, 1983, pp. 258–287.
- [17] J. J. MORE AND D. C. SORESENSEN, *Computing a trust region step*, SIAM J. Sci. Statist. Comput., 4 (1983), pp. 553–572.
- [18] A. M. OSTROWSKI, *Solutions of Equations and Systems of Equations*, Academic Press, New York, 1966.
- [19] B. N. PARLETT, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood Cliffs, NJ, 1980.
- [20] T. PHAM DINH, *Contribution à la théorie de normes and ses applications à l'analyse numérique*, Thèse de Doctorat d'Etat Es Science, Université Joseph-Fourier, Grenoble, 1981.
- [21] T. PHAM DINH, *Convergence of subgradient method for computing the bound norm of matrices*, Linear Algebra Appl., 62 (1984), pp. 163–182.
- [22] T. PHAM DINH, *Algorithmes de calcul du maximum d'une forme quadratique sur la boule unité de la norme du maximum*, Numer. Math., 45 (1984), pp. 377–401.
- [23] T. PHAM DINH AND S. ELBERNOUSSI, *Duality in d.c. (difference of convex functions) optimization. Subgradient methods*, in Trends in Mathematical Optimization, Internat. Ser. Numer. Math. 84, Birkhäuser, Basel, 1988, pp. 276–294.
- [24] T. PHAM DINH, *Méthodes numériques pour la minimisation globale d'une forme quadratique (convexe ou non convexe) sur une boule et une sphère euclidiennes*, Rapport de Recherche, Université Joseph-Fourier, Grenoble, 1989.
- [25] T. PHAM DINH AND S. WANG, *Training multi-layered neural network with a trust region based algorithm*, Math. Modelling Numer. Anal., 24 (1990), pp. 523–553.
- [26] T. PHAM DINH AND L. T. HOAI AN, *Minimisation globale d'une forme quadratique sur une boule and une sphère euclidiennes. Stabilité de la dualité lagrangienne. Optimalité globale. Méthodes numériques*, Rapport de Recherche, L.M.I, CNRS URA 1378, INSA-Rouen, 1992.
- [27] T. PHAM DINH AND L. T. HOAI AN, *Optimisation d.c. (différence de deux fonctions convexes). Dualité et Stabilité. Optimalités locale et globale. Algorithmes de l'optimisation d.c. (DCA)*, Rapport de Recherche, LMI, CNRS URA 1378, INSA-Rouen, 1994.
- [28] T. PHAM DINH AND L. T. HOAI AN, *Stabilité de la dualité lagrangienne en optimisation d.c. (différence de deux fonctions convexes)*, C.R. Acad. Sci. Paris, Ser. I Math., t.318 (1994), pp. 379–384.
- [29] T. PHAM DINH AND L. T. HOAI AN, *Polyhedral d.c. (Difference of Convex Functions) Programming: Theory, Polyhedral d.c. Algorithm (DCA) and Applications*, preprint, LMI, CNRS URA 1378, INSA-Rouen, 1995.
- [30] T. PHAM DINH AND L. T. HOAI AN, *Lagrangian stability and global optimality on nonconvex quadratic minimization over Euclidean balls and spheres*, J. Convex Anal., 2 (1995), pp. 263–276.
- [31] B. POLYAK, *Introduction to Optimization*, Optimization Software, Inc., Publication Division, New York, 1987.

- [32] F. RENDL AND H. WOLKOWICZ, *A Semidefinite Framework to Trust Region Subproblems with Application to Large Scale Minimization*, CORR Report 94-32, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, ON, Canada, 1994.
- [33] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
- [34] R. T. ROCKAFELLAR, *Monotone operators and the proximal point algorithm in convex programming*, SIAM J. Control Optim., 14 (1976), pp. 877–898.
- [35] S. A. SANTOS AND D. C. SORESENSEN, *A new matrix-free algorithm for the large-scale trust-region subproblem*, SIAM J. Optim., submitted.
- [36] D. C. SORESENSEN, *Newton's method with a model trust region modification*, SIAM J. Numer. Anal., 19 (1982), pp. 409–426.
- [37] D. C. SORESENSEN, *Implicit application of polynomial filters in a K-step Arnoldi method*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 357–385.
- [38] D. C. SORESENSEN, *Minimization of a large scale quadratic function subject to a spherical constraint*, SIAM J. Optim., 7 (1997), pp. 141–161.
- [39] J. F. TOLAND, *On subdifferential calculus and duality in nonconvex optimization*, Bull. Soc. Math. France, Mémoire, 60 (1979), pp. 177–183.

AN INCREMENTAL GRADIENT(-PROJECTION) METHOD WITH MOMENTUM TERM AND ADAPTIVE STEPSIZE RULE*

PAUL TSENG†

Abstract. We consider an incremental gradient method with momentum term for minimizing the sum of continuously differentiable functions. This method uses a new adaptive stepsize rule that decreases the stepsize whenever sufficient progress is not made. We show that if the gradients of the functions are bounded and Lipschitz continuous over a certain level set, then every cluster point of the iterates generated by the method is a stationary point. In addition, if the gradient of the functions have a certain growth property, then the method is either linearly convergent in some sense or the stepsizes are bounded away from zero. The new stepsize rule is much in the spirit of heuristic learning rules used in practice for training neural networks via backpropagation. As such, the new stepsize rule may suggest improvements on existing learning rules. Finally, extension of the method and the convergence results to constrained minimization is discussed, as are some implementation issues and numerical experience.

Key words. incremental gradient method, gradient projection, convergence analysis, backpropagation, nonlinear neural network training

AMS subject classifications. 49M07, 49M37, 90C30

PII. S1052623495294797

1. Introduction. Consider the problem of minimizing, over the n -dimensional real space $\mathbb{R}^n$, a function $f : \mathbb{R}^n \mapsto \mathbb{R}$ of the form

$$(1) \quad f(x) = \sum_{i=1}^m f_i(x),$$

where f_i , $i = 1, \dots, m$, are continuously differentiable functions from $\mathbb{R}^n$ to $\mathbb{R}$. Our interest in this problem stems from an important special case, that of nonlinear neural network training, in which x is the vector of weights in the neural network and $f_i(x)$ is the corresponding output error for the i th training example. (See [8], [9], [12], [13] for more detailed discussions of this connection.) Extension of our results to the constrained minimization of f will be discussed in section 5.

We will focus on the following iterative method for solving the preceding problem whereby, for a given $x_1^0 \in \mathbb{R}^m$, we generate a sequence $\{(x_1^t, \dots, x_{m+1}^t)\}_{t=0,1,\dots}$ according to

$$(2) \quad \begin{aligned} x_{i+1}^t &:= x_i^t - \alpha^t d_i^t, \quad i = 1, \dots, m, \\ x_1^{t+1} &:= x_{m+1}^t, \end{aligned}$$

where α^t is a positive scalar (the “stepsize”) and

$$(3) \quad d_i^t := \begin{cases} \nabla f_i(x_i^t) + \zeta d_m^{t-1} & \text{if } i = 1, \\ \nabla f_i(x_i^t) + \zeta d_{i-1}^t & \text{if } i > 1 \end{cases}$$

with $d_m^{-1} = 0$ and $\zeta \in [0, 1)$. Here, each direction d_i^t is a weighted sum of the previous direction (the “momentum term”) and the gradient of f_i at x_i^t . Thus, unlike

*Received by the editors November 15, 1995; accepted for publication (in revised form) January 30, 1997. This work was supported by National Science Foundation grant CCR-9311621.

<http://www.siam.org/journals/siopt/8-2/29479.html>

†Department of Mathematics, University of Washington, Seattle, WA 98195 (tseng@math.washington.edu).

conventional gradient methods, this method does not use the gradient of f to take a step but only the gradient of one of the f_i . In the special case where $m = 1$, this method reduces to the steepest descent method for $\zeta = 0$ and to the heavy-ball method [17, p. 65] for $\zeta \geq 0$. For general m and $\zeta = 0$, this method is reminiscent of a nonlinear least square algorithm of Davidon [5], which has been further studied by Pappas [15] and Bertsekas [2]. When applied to neural network training, this method reduces to the very popular on-line backpropagation algorithm with a momentum term (with “training/learning rates” identified with stepsizes), as conceived by Werbos [20], Le Cun [11], Parker [16], and Rumelhart, Hinton, and Williams [18] (see the discussions in [21]). Numerical experience suggests that it is typically beneficial to choose $\zeta > 0$. (In [18, p. 330], a value of $\zeta \approx .9$ is recommended.) An interesting one-parameter generalization of this method for $\zeta = 0$ and of the steepest descent method is recently studied in [3].

A key issue concerns the stepsizes $\{\alpha^t\}_{t=0,1,\dots}$ which, to quote from [14], “are often crucial for the success of the algorithm.” In the case of neural network training, various heuristic rules for choosing the stepsize have been proposed, the most popular of which entail keeping the stepsize fixed for as long as “progress” is made and decreasing the stepsize if otherwise. However, these heuristic rules are justified only by extensive experimentation (see [10, p. 124], [19], and references therein and in [6]). More recently, stepsize rules have been proposed for the special case of $\zeta = 0$, for which global convergence can be shown under mild assumptions on $f, f_1, \dots, f_m$. One such rule, studied in [3], [8], [12], [13], [22], [23], requires the stepsizes $\{\alpha^t\}_{t=0,1,\dots}$ to be square summable but not summable, i.e.,

$$\sum_{t=0}^{\infty} (\alpha^t)^2 < \infty, \quad \sum_{t=0}^{\infty} \alpha^t = \infty.$$

The reference [8] also considers the more general stepsize rule in which square summability of $\{\alpha^t\}_{t=0,1,\dots}$ is replaced by $\alpha^t \rightarrow 0$. These rules, however, always require the stepsize to tend to zero. To see the drawback of this, suppose $f_1, \dots, f_m$ are identical with Lipschitz continuous gradients. Then the method (2)–(3) with $\zeta = 0$ is just the steepest descent method for minimizing f , for which it is well known that convergence does not require the stepsize to tend to zero (and that a stepsize tending to zero yields slow convergence). A second rule, proposed in [12], requires that

$$\alpha^t \propto \left\| \sum_{i=1}^m \nabla f_i(x_i^{t-1}) \right\|^2.$$

This rule uses information about f , but it still requires the stepsize to tend to zero (the right-hand side tends to zero as $x_1^{t-1}, \dots, x_m^{t-1}$ all tend to a stationary point of f) and in practice the convergence seems to be slow. A third rule, proposed in [9], chooses α^t to be the largest element of $\{\alpha^{t-1}, \omega\alpha^{t-1}, \omega^2\alpha^{t-1}, \dots\}$ for which the following sufficient descent condition is satisfied:

$$f(x_1^{t+1}) \leq f(x_1^t) - \epsilon_1 \alpha^t \left\| \sum_{i=1}^m \nabla f_i(x_i^t) \right\|^2 - \epsilon_2 (\alpha^t)^2 \sum_{i=1}^m \|\nabla f_i(x_i^t)\|^2,$$

where $\omega \in (0, 1)$, $\epsilon_1 \in (0, 1)$, $\epsilon_2 > 0$ are parameters. This rule uses information about f and does not always require the stepsize to tend to zero. Moreover, it is in the spirit of heuristic rules used in practice in that it keeps the stepsize fixed for as long

as sufficient descent is made and decreases the stepsize if otherwise. On the other hand, this rule still tends to make the stepsize very small since it requires sufficient descent at every iteration and when the term $\sum_{i=1}^m \|\nabla f_i(x_i^t)\|^2$ is bounded away from zero, the stepsize must necessarily tend to zero. In addition, the preceding stepsize rules apply to the case of $\zeta = 0$. It is unclear whether these rules can be extended to the case of $\zeta > 0$, which is the case of practical interest. (The work of [13] considers a version of the method that uses a momentum term. However, the momentum term uses only the history of the method from the start of the current iteration.)

In this paper, we propose a new rule (see (5)–(6) and (7)–(9)) for choosing the stepsizes $\{\alpha^t\}_{t=0,1,\dots}$ for which convergence of the method (2)–(3) can be shown for any $\zeta \in [0, (.5)^{1/m}]$. (Note that $(.5)^{1/m} > .9$ for $m \geq 8$, so the restriction on ζ is mild.) This new rule, like the rule proposed in [9] for the case of $\zeta = 0$, does not always require the stepsize to tend to zero and keeps the stepsize fixed for as long as descent in an overall sense is achieved and decreases the stepsize if otherwise. Unlike the rule of [9], this new stepsize rule does not require descent at every iteration and, as such, the stepsize tends to remain large which is essential for good convergence.

We show that the method (2)–(3) using this stepsize rule has desirable global convergence properties (see Proposition 3.4). Moreover, in the case where $\nabla f_1, \dots, \nabla f_m$ grow at most linearly in norm with ∇f (which, in the context of neural network training, amounts to the neural network being trainable so to achieve zero output error on the training examples), either the method is linearly convergent in some sense or the stepsize is bounded away from zero (see Proposition 4.2). The method and the convergence results can also be extended, with suitable modifications, to the problem of constrained minimization of f (see section 5). We note that neither $d_1^t, \dots, d_m^t$ nor their sum need be a descent direction for f , so conventional convergence arguments cannot be applied here. Moreover, for $\zeta > 0$, the proofs are further complicated by the dependence of d_i^t on the entire past history of the method up to then. And while some of our proof ideas are adapted from [12] and [13], much of the arguments are new due to the use of a new stepsize rule and the presence of the momentum term. In section 6, we discuss implementation issues and numerical experience with the method.

A few words about our notations: For any $x, y \in \mathbb{R}^n$, $\langle x, y \rangle$ denotes the usual inner product of vectors x and y and $\|x\|$ denotes the Euclidean norm of x , i.e., $\|x\| = \langle x, x \rangle^{1/2}$. For any continuously differentiable function $h : \mathbb{R}^n \mapsto \mathbb{R}$, we say that ∇h , the gradient of h , is Lipschitz continuous (with constant $\lambda \geq 0$) on a subset X of $\mathbb{R}^n$ if

$$\|\nabla h(x) - \nabla h(x')\| \leq \lambda \|x - x'\| \quad \forall x, x' \in X.$$

For any two positive integers $t > \tau$ and any scalar $\theta > 0$, we denote

$$\theta_\tau^t = \theta^{\tau m} + \theta^{\tau m+1} + \dots + \theta^{tm-1}.$$

Thus, $\theta_s^\tau + \theta_\tau^t = \theta_s^t$ for any $t > \tau > s$. For any $i \in \{1, \dots, m\}$ and positive integer j , we define

$$i \ominus j := (i - j - 1 \bmod m - 1) + 1.$$

Thus $i \ominus j = i - j$ for $j = 1, \dots, i - 1$ and $i \ominus j = i - j + m$ for $j = i, \dots, i + m - 1$ and so on.

2. Method description. In this section, we describe in detail the method (2)–(3) for the unconstrained minimization of f given by (1) and the new rule for choosing the stepsize α^t adaptively. To describe this new stepsize rule, we follow [12] and make the following assumption about $f, f_1, \dots, f_m$ and the initial iterate x_1^0 .

Assumption A. There exist scalars $\eta > f(x_1^0)$ and $\rho > 0$ such that, for $i = 1, \dots, m$, ∇f_i is bounded and Lipschitz continuous (with some constant $\lambda_i \geq 0$) on the set

$$\mathcal{R}_\rho^\eta := \{ x \in \mathbb{R}^n : f(x) \leq \eta \} + \rho \mathcal{B},$$

where $\mathcal{B} = \{ x \in \mathbb{R}^n : \|x\| \leq 1 \}$.

Assumption A is quite mild and, in particular, is satisfied when $f_1, \dots, f_m$ are twice differentiable and the level set $\{ x \in \mathbb{R}^n : f(x) \leq \eta \}$ is bounded for some $\eta > f(x_1^0)$, as is the typical case with neural network training. (See [12, section 3] for further discussions.)

The new stepsize rule for choosing α^t depends on η, ρ , and $\lambda_1, \dots, \lambda_m$ and, in the spirit of the Armijo–Goldstein stepsize rule for gradient descent methods, periodically checks if a certain descent condition is satisfied since the previous check was made and, if not, decreases the stepsize and restarts the method from when the previous check was made. Below, we formally state the method (2)–(3) using this stepsize rule. Following [3], we will call this method the *incremental gradient method*.

Incremental gradient method (with momentum term). Choose any $x_1^0 \in \mathbb{R}^n$ such that Assumption A holds for some η, ρ and $\lambda_1, \dots, \lambda_m$. Choose any $\zeta \in [0, (.5)^{1/m})$ and let

$$(4) \quad \delta_1 := \frac{1 - 2\zeta^m}{1 - \zeta}, \quad \delta_2 := \frac{.5\zeta^m}{1 - \zeta}, \quad \delta_3 := .5(\lambda_1 + \dots + \lambda_m)(1 + \zeta^m).$$

(By choice of ζ , we have $\delta_1 > 0$.) Choose any $\omega \in (0, 1)$ and any subsequence T of $\{1, 2, \dots\}$ containing 1. Choose any positive scalars $\epsilon_0, \epsilon_1, \epsilon_2, \epsilon_3$ satisfying $\epsilon_1 < \delta_1$ and $\epsilon_2 \frac{\zeta^m}{1 - \zeta^m} < \eta - f(x_1^0)$.

Step 0. Let α^0 be the largest element of $\{\epsilon_0, \omega\epsilon_0, \omega^2\epsilon_0, \dots\}$, for which $x_2^0, \dots, x_m^0, x_1^1$ given by (2)–(3) with $t = 0$ satisfy the following two conditions:

$$(5) \quad (x_2^0, \dots, x_m^0, x_1^1) \in (\mathcal{R}_\rho^\eta)^m,$$

$$(6) \quad f(x_1^1) \leq \eta - \delta_3(\alpha^0 \beta^0)^2 - \epsilon_2 \frac{\zeta^m}{1 - \zeta^m},$$

where β^0 is given by (10).

Step 1. For each $s \in T$, let $h = \min\{t \in T : t > s\}$ and we generate $(x_1^h, d_m^{h-1}, \alpha^{h-1})$ from $(x_1^s, d_m^{s-1}, \alpha^{s-1})$ as follows: If $\nabla f(x_1^s) = 0$, we stop. Else, we let α^{h-1} be the largest element of $\{\alpha^{s-1}, \omega\alpha^{s-1}, \omega^2\alpha^{s-1}, \dots\}$ for which $x_2^t, \dots, x_m^t, x_1^{t+1}$ given by (2)–(3) with $\alpha^t = \alpha^{h-1}$, $t = s, \dots, h-1$, satisfy the following three conditions:

$$(7) \quad (x_2^t, \dots, x_m^t, x_1^{t+1}) \in (\mathcal{R}_\rho^\eta)^m, \quad t = s, \dots, h-1,$$

$$(8) \quad f(x_1^h) \leq \eta - (\delta_2 + \epsilon_1)p^h - \delta_3 q^h + \delta_2(1 - \zeta)u^h + \delta_3(1 - \zeta)v^h - \epsilon_2 \frac{\zeta^{hm}}{1 - \zeta^m},$$

$$(9) \quad \|\nabla f(x_1^s) - g^s\| \leq \epsilon_3 \|g^s\|,$$

where, for $t = s, \dots, h-1$, we define

$$(10) \quad g^t := \sum_{i=1}^m \nabla f_i(x_i^t), \quad \beta^t := \max \left\{ \|g^t\|, \sum_{i=1}^m \|d_i^t\| \right\},$$

(11)

$$\begin{aligned} p^{t+1} &:= p^t + \alpha^t \|g^t\|^2, & u^{t+1} &:= u^t + a^t, & a^{t+1} &:= \zeta^m a^t + (1 + \dots + \zeta^{m-1}) \alpha^t \|g^t\|^2, \\ q^{t+1} &:= q^t + (\alpha^t \beta^t)^2, & v^{t+1} &:= v^t + b^t, & b^{t+1} &:= \zeta^m b^t + (1 + \dots + \zeta^{m-1}) (\alpha^t \beta^t)^2, \end{aligned}$$

with $p^1 := u^1 := a^1 := q^0 := v^0 := b^0 := 0$.

Roughly speaking, the stepsize rule checks at each iteration $s \in T$ whether the current stepsize is acceptable (i.e., satisfies (7)–(9)) for all iterations between s and the next element h of T and, if not, it decreases the stepsize by ω and restarts the method from iteration s . Thus, if the elements of T are far apart, then the rule makes this check infrequently but it needs to backtrack further whenever the stepsize needs to be decreased. In our testing (see section 6), checking every 10 iterations, i.e., $T = \{1, 11, 21, 31, \dots\}$, worked well. A more sophisticated strategy might be to check more frequently at the beginning, e.g., $T = \{1, 2, 5, 11, 21, 31, \dots\}$. We remark that we can also increase the stepsize, provided that this is done only a finite number of times. For the other parameters in the stepsize rule, it suffices to choose ϵ_1, ϵ_2 reasonably small, choose ϵ_3, η, ρ reasonably large, choose ϵ_0 near 1, and choose ω, ζ away from 0 and 1. Only the parameter $\lambda_1 + \dots + \lambda_m$, which is problem dependent, requires significant fine tuning (if this is too large, then the stepsize becomes too small; if this is too small, then the stepsize remains large but the method experiences large oscillations). In our testing (see section 6), the choice of $\epsilon_1 = \epsilon_2 = .00001$, $\epsilon_3 = 1000$, $\eta = 1.5f(x_1^0) + 100$, $\rho = \infty$, $\epsilon_0 = 1$, $\omega = .5$, $\zeta = .8$, and $\lambda_1 + \dots + \lambda_m = 1$ worked well. However, to solve a wider range of problems, we would need to update λ_i on-line by, for example, $\|\nabla f_i(x_i^1) - \nabla f_i(x_i^0)\|/\|x_i^1 - x_i^0\|$, for $i = 1, \dots, m$.

Of the three conditions (7)–(9), both (7) and (9) are quite unrestrictive since ρ is typically large (e.g., in [8] and [13], it is assumed that $\rho = \infty$) and we can choose ϵ_3 arbitrarily large. To see that (8) is also unrestrictive, first note that (8) does not require f -value to be monotonically decreasing in any sense but only that the f -value be less than η by some positive quantity. This positive quantity, which is easily computable using the updating formula (11), depends on $f, f_1, \dots, f_m$ and is increasing with t only in a long-run sense. More precisely, by using a straightforward calculation from (11), we see that

(12)

$$p^t = \sum_{\tau=1}^{t-1} \alpha^\tau \|g^\tau\|^2, \quad q^t = \sum_{\tau=0}^{t-1} (\alpha^\tau \beta^\tau)^2, \quad u^t = \sum_{\tau=1}^{t-2} \zeta_0^{t-\tau-1} \alpha^\tau \|g^\tau\|^2, \quad v^t = \sum_{\tau=0}^{t-2} \zeta_0^{t-\tau-1} (\alpha^\tau \beta^\tau)^2.$$

Since $\zeta_0^{t-\tau-1} \approx 1/(1-\zeta)$ for $\tau = 0, \dots, t-2$ (this is true especially for large m), then $p^t \approx (1-\zeta)u^{t+1}$ and $q^t \approx (1-\zeta)v^{t+1}$, so the right-hand side of (8) may increase or decrease with h , depending on how $\alpha^\tau \|g^\tau\|^2$ and $(\alpha^\tau \beta^\tau)^2$ change with τ , though in the long run the tendency is towards a decrease. To see this, we note from (12) and $\zeta_0^{t-\tau-1} \leq 1/(1-\zeta)$ for $\tau = 0, \dots, t-2$ that (8) implies

$$(13) \quad f(x_1^h) \leq \eta - \epsilon_1 \sum_{\tau=1}^{h-1} \alpha^\tau \|g^\tau\|^2 - \delta_2 \alpha^{h-1} \|g^{h-1}\|^2 - \delta_3 (\alpha^{h-1} \beta^{h-1})^2 - \epsilon_2 \frac{\zeta^{hm}}{1-\zeta^m}.$$

The second term on the right-hand side, which is the dominant term there, is decreasing with h .

3. Global convergence analysis. In this section we show that the incremental gradient method of section 2 has desirable global convergence properties (see Proposition 3.4). Throughout, we will assume that Assumption A holds. First, we have the following technical lemma.

LEMMA 3.1. *For any $t \in \{0, 1, \dots\}$, any $\alpha^t > 0$ and any $x_1^t, \dots, x_{m+1}^t$ in $\mathcal{R}_\rho^\eta$ satisfying (2), we have*

$$\|\nabla f_i(x_j^t) - \nabla f_i(x_i^t)\| \leq \lambda_i \alpha^t \beta^t,$$

for $1 \leq i, j \leq m+1$, where β^t is given by (10).

Proof. By (2) and (10), for any $1 \leq j \leq i \leq m+1$, we have

$$\|x_j^t - x_i^t\| = \alpha^t \left\| \sum_{l=j}^{i-1} d_l^t \right\| \leq \alpha^t \beta^t.$$

A similar argument shows the above inequality also holds for any $1 \leq i \leq j \leq m+1$. Since $x_1^t, \dots, x_{m+1}^t$ are in $\mathcal{R}_\rho^\eta$, the above inequality, together with ∇f_i being Lipschitz continuous (with constant λ_i) on $\mathcal{R}_\rho^\eta$ for $i = 1, \dots, m$, yields the desired inequality. $\square$

Under Assumption A, there exist positive scalars $\beta_1, \dots, \beta_m$ such that

$$(14) \quad \|\nabla f_i(x)\| \leq \beta_i \quad \forall x \in \mathcal{R}_\rho^\eta, \quad i = 1, \dots, m.$$

Let

$$(15) \quad \beta := \beta_1 + \dots + \beta_m, \quad \lambda := \lambda_1 + \dots + \lambda_m.$$

The next lemma shows that, for α^t sufficiently small, $x_1^t, \dots, x_{m+1}^t$ satisfying (2)–(3) remain in $\mathcal{R}_\rho^\eta$ for $t = 0, 1, \dots$

LEMMA 3.2. *For any $t \in \{0, 1, \dots\}$, any $\alpha^\tau \in (0, \rho(1-\zeta)/\beta]$ and any $x_1^\tau, \dots, x_{m+1}^\tau$ satisfying (2)–(3) for $\tau = 0, 1, \dots, t$, and such that $(x_1^\tau, \dots, x_m^\tau) \in (\mathcal{R}_\rho^\eta)^m$ for $\tau = 0, 1, \dots, t-1$ and $f(x_1^t) \leq \eta$, we have that $x_1^t, \dots, x_m^t$ are in $\mathcal{R}_\rho^\eta$, as is the line segment joining x_1^t with x_1^{t+1} .*

Proof. First, we claim that

$$(16) \quad \|x_l^t - x_1^t\| \leq \frac{\rho(1-\zeta)}{\beta} \sum_{j=1}^{l-1} \sum_{k=0}^{tm+j-1} \zeta^k \beta_{j \ominus k},$$

for $l = 1, \dots, m+1$. We prove this by induction on l . Clearly, (16) holds for $l = 1$. Suppose (16) holds for $l = 1, \dots, i$ for some $i \in \{1, \dots, m\}$ and we show below that it also holds for $l = i+1$. For $l = 1, \dots, i$, since (16) holds and the right-hand side of (16) is bounded above by ρ , we have $\|x_l^t - x_1^t\| \leq \rho$ so that (cf. $f(x_1^t) \leq \eta$) $x_l^t \in \mathcal{R}_\rho^\eta$. Then, (2)–(3) yields

$$\begin{aligned} \|x_{i+1}^t - x_1^t\| &= \alpha^t \left\| \sum_{k=0}^{tm+i-1} \zeta^k \nabla f_{i \ominus k} \left(x_{i \ominus k}^{\lfloor \frac{tm+i-1-k}{m} \rfloor} \right) \right\| \leq \alpha^t \sum_{k=0}^{tm+i-1} \zeta^k \beta_{i \ominus k} \\ &\leq \frac{\rho(1-\zeta)}{\beta} \sum_{k=0}^{tm+i-1} \zeta^k \beta_{i \ominus k}, \end{aligned}$$

where the first inequality follows from $x_1^0, x_2^0, \dots, x_{i-1}^t, x_i^t \in \mathcal{R}_\rho^\eta$ and (14). Since (16) holds for $l = i$, this shows that (16) holds for $l = i + 1$.

Since (16) holds and the right-hand side of (16) is bounded above by ρ , $f(x_1^t) \leq \eta$ implies $x_l^t \in \mathcal{R}_\rho^\eta$ for $l = 1, \dots, m + 1$. Moreover, (16) with $l = m + 1$ and (2) implies $\|x_1^{t+1} - x_1^t\| \leq \rho$, so the line segment joining x_1^t with x_1^{t+1} lies in $\mathcal{R}_\rho^\eta$. $\square$

By using Lemma 3.1, we obtain the following lemma estimating the decrease in f -value per iteration of the incremental gradient method.

LEMMA 3.3. *For any $t \in \{1, 2, \dots\}$, any $\alpha^\tau > 0$ and any $x_1^\tau, \dots, x_{m+1}^\tau$ in $\mathcal{R}_\rho^\eta$ satisfying (2)–(3) for $\tau = 0, 1, \dots, t$, and such that the line segment joining x_1^t with x_1^{t+1} lies in $\mathcal{R}_\rho^\eta$, we have*

$$(17) \quad \begin{aligned} f(x_1^{t+1}) &\leq f(x_1^t) - (\delta_1 + \delta_2)\alpha^t\|g^t\|^2 + \lambda(1.5 + 2\zeta\zeta_0^\infty)(\alpha^t\beta^t)^2 \\ &\quad + \delta_2(1 - \zeta) \sum_{\tau=1}^{t-1} \zeta_{t-\tau-1}^{t-\tau} \alpha^t\|g^\tau\|^2 + \delta_3(1 - \zeta) \sum_{\tau=0}^{t-1} \zeta_{t-\tau-1}^{t-\tau} (\alpha^\tau\beta^\tau)^2 + \alpha^t\|g^t\|\zeta^{tm}\bar{\beta}, \end{aligned}$$

where $\delta_1, \delta_2, \delta_3$ are given by (4), g^t, β^t are given by (10), and $\bar{\beta} := \|\sum_{j=1}^m \sum_{i=j}^m \zeta^{i-j} \nabla f_j(x_j^0)\|$. Similarly, for any $\alpha^0 > 0$ and any $x_1^0, \dots, x_m^0, x_1^1$ in $\mathcal{R}_\rho^\eta$ satisfying (2)–(3) (with $t = 0$) and such that the line segment joining x_1^0 with x_1^1 lies in $\mathcal{R}_\rho^\eta$, we have

$$(18) \quad f(x_1^1) \leq f(x_1^0) - \alpha^0\|g^0\|^2 + 1.5\lambda(\alpha^0\beta^0)^2 + 2\alpha^0\|g^0\|\beta^0.$$

Proof. Fix any $t \in \{1, 2, \dots\}$. Since the line segment joining x_1^t with x_1^{t+1} lies in $\mathcal{R}_\rho^\eta$ and, by $\nabla f = \nabla f_1 + \dots + \nabla f_m$ (cf. (1)) and Assumption A, ∇f is Lipschitz continuous (with constant λ given by (15)) on $\mathcal{R}_\rho^\eta$, we obtain from the intermediate value theorem (see [4, p. 639]) that

$$(19) \quad f(x_1^{t+1}) \leq f(x_1^t) + \langle \nabla f(x_1^t), x_1^{t+1} - x_1^t \rangle + .5\lambda\|x_1^{t+1} - x_1^t\|^2.$$

Using (2) and (10), we bound the rightmost term in (19) as follows:

$$(20) \quad \|x_1^{t+1} - x_1^t\| = \alpha^t \left\| \sum_{i=1}^m d_i^t \right\| \leq \alpha^t \beta^t.$$

The second term on the right-hand side of (19) can be bounded as follows:

$$(21) \quad \begin{aligned} &\langle \nabla f(x_1^t), x_1^{t+1} - x_1^t \rangle \\ &= \langle \nabla f(x_1^t) - g^t, x_1^{t+1} - x_1^t \rangle + \left\langle g^t, -\alpha^t \sum_{i=1}^m d_i^t + \alpha^t \zeta_0^1 g^t \right\rangle - \alpha^t \zeta_0^1 \|g^t\|^2 \\ &\leq \|\nabla f(x_1^t) - g^t\| \|x_1^{t+1} - x_1^t\| + \alpha^t \|g^t\| \left\| \sum_{i=1}^m d_i^t - \zeta_0^1 g^t \right\| - \alpha^t \zeta_0^1 \|g^t\|^2, \end{aligned}$$

where the equality follows from (2). Also, we have from (3) and (10) that

$$\begin{aligned}
& \left\| \sum_{i=1}^m d_i^t - \zeta_0^1 g^t \right\| \\
&= \left\| \sum_{i=1}^m \left(\sum_{\tau=0}^{t-1} \left(\sum_{j=m-i+1}^m \zeta^{(t-\tau)m-j} \nabla f_{i+j-m}(x_{i+j-m}^{\tau+1}) \right. \right. \right. \\
&\quad \left. \left. + \sum_{j=1}^{m-i} \zeta^{(t-\tau)m-j} \nabla f_{i+j}(x_{i+j}^{\tau}) \right) + \sum_{j=1}^i \zeta^{tm+i-j} \nabla f_j(x_j^0) \right) - \zeta_0^1 g^t \right\| \\
&= \left\| \sum_{\tau=0}^{t-1} \left(\sum_{j=1}^m \zeta^{(t-\tau)m-j} \left(\sum_{k=1}^j \nabla f_k(x_k^{\tau+1}) + \sum_{k=j+1}^m \nabla f_k(x_k^{\tau}) \right) \right) \right. \\
&\quad \left. + \sum_{j=1}^m \sum_{i=j}^m \zeta^{tm+i-j} \nabla f_j(x_j^0) - \zeta_0^1 g^t \right\| \\
&= \left\| \sum_{\tau=0}^{t-1} \sum_{j=1}^m \zeta^{(t-\tau)m-j} g_j^{\tau+1} + \sum_{j=1}^m \sum_{i=j}^m \zeta^{tm+i-j} \nabla f_j(x_j^0) - \sum_{j=1}^m \zeta^{m-j} g^t \right\| \\
&= \left\| \sum_{\tau=0}^{t-2} \sum_{j=1}^m \zeta^{(t-\tau)m-j} g_j^{\tau+1} + \sum_{j=1}^m \sum_{i=j}^m \zeta^{tm+i-j} \nabla f_j(x_j^0) + \sum_{j=1}^m \zeta^{m-j} (g_j^t - g^t) \right\| \\
&\leq \sum_{\tau=0}^{t-2} \sum_{j=1}^m \zeta^{(t-\tau)m-j} (\|g^{\tau+1}\| + \|g_j^{\tau+1} - g^{\tau+1}\|) + \left\| \sum_{j=1}^m \sum_{i=j}^m \zeta^{tm+i-j} \nabla f_j(x_j^0) \right\| \\
&\quad + \sum_{j=1}^m \zeta^{m-j} \|g_j^t - g^t\| \\
&= \sum_{\tau=0}^{t-2} \sum_{j=1}^m \zeta^{(t-\tau)m-j} \|g^{\tau+1}\| + \sum_{\tau=0}^{t-1} \sum_{j=1}^{m-1} \zeta^{(t-\tau)m-j} \|g_j^{\tau+1} - g^{\tau+1}\| + \zeta^{tm} \bar{\beta} \\
&\leq \sum_{\tau=0}^{t-2} \sum_{j=1}^m \zeta^{(t-\tau)m-j} \|g^{\tau+1}\| + \lambda \sum_{\tau=0}^{t-1} \sum_{j=0}^{m-1} \zeta^{(t-\tau)m-j} (\alpha^{\tau+1} \beta^{\tau+1} + \alpha^{\tau} \beta^{\tau}) + \zeta^{tm} \bar{\beta} \\
(22) \quad &= \sum_{\tau=1}^{t-1} \zeta_{t-\tau}^{t-\tau+1} \|g^{\tau}\| + \lambda \zeta \left(\sum_{\tau=0}^{t-1} \zeta_{t-\tau-1}^{t-\tau+1} \alpha^{\tau} \beta^{\tau} + \zeta_0^1 \alpha^t \beta^t - \zeta_{t-1}^t \alpha^0 \beta^0 \right) + \zeta^{tm} \bar{\beta},
\end{aligned}$$

where the second equality follows from interchanging the order of the summations over i and j and then making the substitution $k = i + j - m$ (respectively, $k = i + j$) in the first (respectively, second) summation inside the doubly nested parentheses; the third equality follows by letting

$$(23) \quad g_j^{\tau+1} := \sum_{k=1}^j \nabla f_k(x_k^{\tau+1}) + \sum_{k=j+1}^m \nabla f_k(x_k^{\tau});$$

the fifth equality uses the fact $g_m^{\tau+1} = g^{\tau+1}$; the second inequality follows from (15) and the following consequence of (10) and Lemma 3.1 (since $x_1^\tau, \dots, x_{m+1}^\tau = x_1^{\tau+1}, \dots, x_m^{\tau+1} \in \mathcal{R}_\rho^\eta$):

$$\begin{aligned} \|g_j^{\tau+1} - g^{\tau+1}\| &= \left\| \sum_{k=j+1}^m (\nabla f_k(x_k^\tau) - \nabla f_k(x_{m+1}^\tau) + \nabla f_k(x_1^{\tau+1}) - \nabla f_k(x_k^{\tau+1})) \right\| \\ &\leq \sum_{k=j+1}^m \lambda_k (\alpha^\tau \beta^\tau + \alpha^{\tau+1} \beta^{\tau+1}) \end{aligned}$$

for $j = 1, \dots, m-1$ and for $\tau = 0, 1, \dots, t-1$. By a similar argument, we have that

$$(24) \quad \|\nabla f(x_1^t) - g^t\| = \left\| \sum_{i=1}^m (\nabla f_i(x_1^t) - \nabla f_i(x_i^t)) \right\| \leq \sum_{i=1}^m \lambda_i \alpha^t \beta^t = \lambda \alpha^t \beta^t.$$

Using (20)–(24) to bound the right-hand side of (19) and then using β^t to bound $\|g^t\|$ (cf. (10)) yields

$$\begin{aligned} f(x_1^{t+1}) &\leq f(x_1^t) - \zeta_0^1 \alpha^t \|g^t\|^2 + 1.5\lambda(\alpha^t \beta^t)^2 + \alpha^t \|g^t\| \sum_{\tau=1}^{t-1} \zeta_{t-\tau}^{t-\tau+1} \|g^\tau\| \\ &\quad + \alpha^t \beta^t \lambda \zeta \left(\sum_{\tau=0}^{t-1} \zeta_{t-\tau-1}^{t-\tau+1} \alpha^\tau \beta^\tau + \zeta_0^1 \alpha^t \beta^t \right) + \alpha^t \|g^t\| \zeta^{tm} \bar{\beta} \\ &\leq f(x_1^t) - \zeta_0^1 \alpha^t \|g^t\|^2 + 1.5\lambda(\alpha^t \beta^t)^2 + .5\alpha^t \left(\zeta_1^t \|g^t\|^2 + \sum_{\tau=1}^{t-1} \zeta_{t-\tau}^{t-\tau+1} \|g^\tau\|^2 \right) \\ &\quad + \lambda \zeta \left(.5 \left((\zeta_0^t + \zeta_1^{t+1})(\alpha^t \beta^t)^2 + \sum_{\tau=0}^{t-1} \zeta_{t-\tau-1}^{t-\tau+1} (\alpha^\tau \beta^\tau)^2 \right) \right. \\ &\quad \left. + \zeta_0^1 (\alpha^t \beta^t)^2 \right) + \alpha^t \|g^t\| \zeta^{tm} \bar{\beta} \\ &= f(x_1^t) - (\zeta_0^1 - .5\zeta_1^t) \alpha^t \|g^t\|^2 + \lambda(1.5 + .5\zeta\zeta_0^t + .5\zeta\zeta_1^{t+1} + \zeta\zeta_0^1)(\alpha^t \beta^t)^2 \\ &\quad + .5\alpha^t \zeta^m \sum_{\tau=1}^{t-1} \zeta_{t-\tau-1}^{t-\tau} \|g^\tau\|^2 + .5\lambda\zeta(1 + \zeta^m) \sum_{\tau=0}^{t-1} \zeta_{t-\tau-1}^{t-\tau} (\alpha^\tau \beta^\tau)^2 + \alpha^t \|g^t\| \zeta^{tm} \bar{\beta}, \end{aligned}$$

where the second inequality follows from using $ab \leq .5(a^2 + b^2)$ and the equality uses properties of ζ_i^j . Bounding from above ζ_1^t by $\zeta_1^\infty = \zeta^m/(1 - \zeta)$ and $\zeta_0^t, \zeta_1^{t+1}, \zeta_0^1$ by ζ_0^∞ in the above expression and then using (4) yields (17). The proof of (18) is very similar. $\square$

Below, we state and prove the main global convergence result for the incremental gradient method of section 2. The proof uses Lemmas 3.1–3.3 to show that the method is well defined and uses the observation (13) to show that (29) holds (unless $\nabla f(x_1^s) = 0$ for some $s \in T$ or $\liminf_{t \rightarrow \infty} f(x_1^t) = -\infty$). The latter in turn is used to show that $\{g^t\}_{t=0,1,\dots} \rightarrow 0$.

PROPOSITION 3.4. *The sequences $\{(x_1^t, \dots, x_{m+1}^t)\}_{t=0,1,\dots}$ and $\{\alpha^t\}_{t=0,1,\dots}$ generated by the incremental gradient method (see (2)–(3) and (4)–(11)) are well defined.*

Moreover, either (i) $\nabla f(x_1^s) = 0$ for some $s \in T$ or (ii) $\liminf_{t \rightarrow \infty} f(x_1^t) = -\infty$ or (iii) $\{g^t\}_{t=0,1,\dots} \rightarrow 0$ and $\{\nabla f(x_1^s)\}_{s \in T} \rightarrow 0$, where T is the subsequence of $\{1, 2, \dots\}$ specified in the method.

Proof. We will show by induction on t that α^t is well defined for $t = 0, 1, \dots$. Since $f(x_1^0) < \eta$, for $\alpha^0 \leq \rho(1 - \zeta)/\beta$, Lemma 3.2 shows that $x_1^0, \dots, x_m^0$ are in $\mathcal{R}_\rho^\eta$ as is the line segment joining x_1^0 with x_1^1 . Thus (5) holds and, by Lemma 3.3, (18) holds. Since $\epsilon_2 \frac{\zeta^m}{1 - \zeta^m} < \eta - f(x_1^0)$, the latter implies that (6) holds for all α^0 sufficiently small, so α^0 , being the largest element of $\{\epsilon_0, \omega\epsilon_0, \omega^2\epsilon_0, \dots\}$ for which (6) holds, is well defined. Now assume that, for some $s \in T$, α^t is well defined for $t = 0, 1, \dots, s - 1$. If $\nabla f(x_1^s) = 0$, then we stop with case (i). Suppose instead that $\nabla f(x_1^s) \neq 0$ and let $h = \min\{t \in T : t > s\}$. We argue in the next two paragraphs that α^t is well defined for $t = s, \dots, h - 1$.

Since $g^t \rightarrow \nabla f(x_1^s)$ and β^t is bounded as $\alpha^t \rightarrow 0$ for $t = s, \dots, h - 1$, we have

$$(25) \quad \begin{aligned} \alpha^t &\leq \frac{\rho(1 - \zeta)}{\beta}, \quad \alpha^t \leq \frac{(\delta_1 - \epsilon_1)\|g^t\|^2}{(\lambda(1.5 + 2\zeta\zeta_0^\infty) + \delta_3)(\beta^t)^2}, \\ \alpha^t &\leq \frac{\epsilon_2}{\|g^t\|\beta}, \quad \alpha^s \leq \frac{\epsilon_3\|g^s\|}{\lambda\beta^s}, \quad t = s, \dots, h - 1, \end{aligned}$$

for any $\alpha^s = \dots = \alpha^{h-1}$ sufficiently small. We claim that, whenever (25) holds, then

$$(26) \quad f(x_1^{t+1}) \leq \eta, \quad (x_2^t, \dots, x_m^t, x_1^{t+1}) \in (\mathcal{R}_\rho^\eta)^m,$$

for $t = s - 1, \dots, h - 1$. Clearly (26) holds for $t = s - 1$. (This follows from (5)–(6) when $s = 1$ and follows from (7) with $t = s - 1$ and (13) with $h = s$, with the latter implied by (8) with $h = s$, when $s > 1$.) Suppose that, for some $k \in \{s, \dots, h - 1\}$, the relation (26) holds for $t = s - 1, \dots, k - 1$. Then, $(x_1^t, \dots, x_m^t) \in (\mathcal{R}_\rho^\eta)^m$ for $t = 0, 1, \dots, k - 1$ (since (26) holds for $t = s - 1, \dots, k - 1$ and (7) holds for $t = 0, 1, \dots, s - 1$ and $f(x_1^0) < \eta$) and, for each $t \in \{s, \dots, k\}$, $f(x_1^t) \leq \eta$. Since we also have $\alpha^t \leq \rho(1 - \zeta)/\beta$ (by (25)) for each $t \in \{s, \dots, k\}$, Lemma 3.2 then yields that $x_1^t, \dots, x_m^t$ and the line segment joining x_1^t with x_1^{t+1} all lie in $\mathcal{R}_\rho^\eta$, and, by Lemma 3.3, (17) holds. Using the second and third inequalities in (25) together with the fact $\alpha^t \leq \alpha^\tau$, for $\tau = 0, 1, \dots, t - 1$, to bound the right-hand side of (17) yields

$$(27) \quad \begin{aligned} f(x_1^{t+1}) &\leq f(x_1^t) - (\delta_2 + \epsilon_1)\alpha^t\|g^t\|^2 - \delta_3(\alpha^t\beta^t)^2 + \delta_2(1 - \zeta) \sum_{\tau=1}^{t-1} \zeta_{t-\tau-1}^{t-\tau} \alpha^\tau\|g^\tau\|^2 \\ &\quad + \delta_3(1 - \zeta) \sum_{\tau=0}^{t-1} \zeta_{t-\tau-1}^{t-\tau} (\alpha^\tau\beta^\tau)^2 + \epsilon_2\zeta^{tm} \end{aligned}$$

for $t = s, \dots, k$. Also, we have that the inequality (8) holds for $h = s$. (For $s = 1$, this follows from (6); for $s > 1$, this follows from α^{s-1} being chosen in Step 1 such that (8) holds with $h = s$.) By using (12), we can rewrite this inequality equivalently as

$$\begin{aligned} f(x_1^s) &\leq \eta - (\delta_2 + \epsilon_1) \sum_{\tau=1}^{s-1} \alpha^\tau\|g^\tau\|^2 - \delta_3 \sum_{\tau=0}^{s-1} (\alpha^\tau\beta^\tau)^2 + \delta_2(1 - \zeta) \sum_{\tau=1}^{s-2} \zeta_0^{s-\tau-1} \alpha^\tau\|g^\tau\|^2 \\ &\quad + \delta_3(1 - \zeta) \sum_{\tau=0}^{s-2} \zeta_0^{s-\tau-1} (\alpha^\tau\beta^\tau)^2 - \frac{\epsilon_2\zeta^{sm}}{1 - \zeta^m}. \end{aligned}$$

Summing (27) over all $t \in \{s, \dots, k\}$ and then adding to it the above inequality, we obtain

$$\begin{aligned}
f(x_1^{k+1}) &\leq \eta - (\delta_2 + \epsilon_1) \sum_{\tau=1}^k \alpha^\tau \|g^\tau\|^2 - \delta_3 \sum_{\tau=0}^k (\alpha^\tau \beta^\tau)^2 - \epsilon_2 \left(\frac{\zeta^{sm}}{1 - \zeta^m} - \sum_{t=s}^k \zeta^{tm} \right) \\
&\quad + \delta_2(1 - \zeta) \left(\sum_{\tau=1}^{s-2} \zeta_0^{s-\tau-1} \alpha^\tau \|g^\tau\|^2 + \sum_{t=s}^k \sum_{\tau=1}^{t-1} \zeta_{t-\tau-1}^{t-\tau} \alpha^\tau \|g^\tau\|^2 \right) \\
&\quad + \delta_3(1 - \zeta) \left(\sum_{\tau=0}^{s-2} \zeta_0^{s-\tau-1} (\alpha^\tau \beta^\tau)^2 + \sum_{t=s}^k \sum_{\tau=0}^{t-1} \zeta_{t-\tau-1}^{t-\tau} (\alpha^\tau \beta^\tau)^2 \right) \\
&= \eta - (\delta_2 + \epsilon_1) \sum_{\tau=1}^k \alpha^\tau \|g^\tau\|^2 - \delta_3 \sum_{\tau=0}^k (\alpha^\tau \beta^\tau)^2 - \epsilon_2 \frac{\zeta^{(k+1)m}}{1 - \zeta^m} \\
(28) \quad &+ \delta_2(1 - \zeta) \sum_{\tau=1}^{k-1} \zeta_0^{k-\tau} \alpha^\tau \|g^\tau\|^2 + \delta_3(1 - \zeta) \sum_{\tau=0}^{k-1} \zeta_0^{k-\tau} (\alpha^\tau \beta^\tau)^2 \\
&\leq \eta - (\delta_2 + \epsilon_1) \sum_{\tau=1}^k \alpha^\tau \|g^\tau\|^2 - \delta_3 \sum_{\tau=0}^k (\alpha^\tau \beta^\tau)^2 + \delta_2 \sum_{\tau=1}^{k-1} \alpha^\tau \|g^\tau\|^2 + \delta_3 \sum_{\tau=0}^{k-1} (\alpha^\tau \beta^\tau)^2,
\end{aligned}$$

where the equality follows from exchanging the order of summation and using the definition of ζ_i^j ; the last inequality follows from the facts $\zeta_0^j \leq \zeta_0^\infty = 1/(1 - \zeta)$ for all $j > 0$. The right-hand side of the above inequality is less than η , so the claim (26) holds for $t = k$. By induction on k , it follows that (26) holds for $t = s - 1, \dots, h - 1$.

Thus, if (25) holds, then (26) holds for $t = s - 1, \dots, h - 1$, so that (7) holds. In addition, our argument showed that (28) holds for $k = s, \dots, h - 1$, so that, upon letting $k = h - 1$ in (28) and using (12), we obtain (8). Also, we have from (24) with $t = s$ and the last inequality in (25) that

$$\|\nabla f(x_1^s) - g^s\| \leq \lambda \alpha^s \beta^s \leq \epsilon_3 \|g^s\|,$$

so (9) holds. Thus (7)–(9) hold whenever $\alpha^s = \dots = \alpha^{h-1}$ are sufficiently small. Then, $\alpha^s, \dots, \alpha^{h-1}$, being the largest element of $\{\alpha^{s-1}, \omega \alpha^{s-1}, \dots\}$ such that (7)–(9) hold, are well defined. This completes the induction step and shows that α^t is well defined for all $t = 0, 1, \dots$

There are three cases: either (i) $\nabla f(x_1^s) = 0$ for some $s \in T$ or else, since (8) holds and hence (13) holds for all $h \in T$, (ii) $\liminf_{t \rightarrow \infty} f(x_1^t) = -\infty$, or (iii)

$$(29) \quad \sum_{t=1}^{\infty} \alpha^t \|g^t\|^2 < \infty.$$

Also, we have from $f(x_1^0) < \eta$ and (5) and (7) (with $h = \min\{t \in T : t > s\}$) for all $s \in T$ that

$$(30) \quad (x_1^t, \dots, x_m^t) \in (\mathcal{R}_\rho^\eta)^m, \quad t = 0, 1, \dots$$

We claim that in case (iii), $\{g^t\} \rightarrow 0$. Since $\{\alpha^t\}$ is monotonically decreasing, either $\liminf_{t \rightarrow \infty} \alpha^t > 0$ or $\{\alpha^t\} \downarrow 0$. In the first case, (29) yields $\{g^t\} \rightarrow 0$. Consider now the second case. We showed earlier that, for each $s \in T$, (7)–(9) hold whenever

(25) holds, where $h = \min\{t \in T : t > s\}$. Then the choice of $\alpha^s = \dots = \alpha^{h-1}$ being the largest element of $\{\alpha^{s-1}, \omega\alpha^{s-1}, \dots\}$ for which (7)–(9) hold, implies either $\alpha^s = \alpha^{s-1}$ or

$$(31) \quad \frac{\alpha^s}{\omega} > \min_{t=s, \dots, h-1} \left\{ \frac{\rho(1-\zeta)}{\beta}, \frac{(\delta_1 - \epsilon_1)\|g^t\|^2}{(\lambda(1.5 + 2\zeta\zeta_0^\infty) + \delta_3)(\beta^t)^2}, \frac{\epsilon_2}{\|g^t\|\bar{\beta}}, \frac{\epsilon_3\|g^s\|}{\lambda\beta^s} \right\}.$$

Since (30) holds, then for $t = 0, 1, \dots$ we have from (10) and (14)–(15) that

$$\|g^t\| = \left\| \sum_{i=1}^m \nabla f_i(x_i^t) \right\| \leq \sum_{i=1}^m \beta_i = \beta$$

and from (3) and (14)–(15) that

$$\sum_{i=1}^m \|d_i^t\| = \sum_{i=1}^m \left\| \sum_{k=0}^{tm+i-1} \zeta^k \nabla f_{i \ominus k} \left(x_{i \ominus k}^{\lfloor \frac{tm+i-1-k}{m} \rfloor} \right) \right\| \leq \sum_{i=1}^m \sum_{k=0}^{tm+i-1} \zeta^k \beta_{i \ominus k} \leq \frac{\beta}{1-\zeta},$$

so (10) yields $\beta^t \leq \beta/(1-\zeta)$. Since $\{\alpha^s\} \downarrow 0$ so that (31) holds for all s in some subsequence of T , it follows that $g^t \rightarrow 0$ for t along some subsequence of $\{0, 1, \dots\}$. We now argue that $g^t \rightarrow 0$ for t along the entire sequence $\{0, 1, \dots\}$. Suppose this is not the case so there exists $\epsilon > 0$ such that $\|g^t\| > \epsilon$ for all t along some subsequence of $\{0, 1, \dots\}$. The following argument is a modification of the proof of [13, Theorem 2.1]. Consider any t such that $\|g^t\| \geq \epsilon$. Since $\{\|g^t\|\}_{t=0,1,\dots}$ contains a subsequence that tends to zero, there exists a smallest integer $t' > t$ such that $\|g^{t'}\| < \epsilon/2$. Then,

$$\begin{aligned} \frac{\epsilon}{2} &\leq \|g^t\| - \|g^{t'}\| \\ &\leq \|g^t - g^{t'}\| \\ &= \left\| \sum_{\tau=t}^{t'-1} \sum_{i=1}^m (\nabla f_i(x_i^{\tau+1}) - \nabla f_i(x_1^{\tau+1}) + \nabla f_i(x_{m+1}^{\tau}) - \nabla f_i(x_i^{\tau})) \right\| \\ &\leq \sum_{\tau=t}^{t'-1} \sum_{i=1}^m \lambda_i (\alpha^{\tau+1} \beta^{\tau+1} + \alpha^{\tau} \beta^{\tau}) \\ &\leq \frac{2\lambda\beta}{1-\zeta} \sum_{\tau=t}^{t'-1} \alpha^{\tau}, \end{aligned}$$

where the equality uses (2) and (10); the fourth inequality follows from (30) and Lemma 3.1; the last inequality follows from (15), $\beta^{\tau} \leq \beta/(1-\zeta)$ for all τ and the monotone decreasing property of $\{\alpha^{\tau}\}_{\tau=0,1,\dots}$. We also have that $\|g^{\tau}\| \geq \epsilon/2$ for $\tau = t, \dots, t' - 1$, which together with the above relation yield

$$\sum_{\tau=t}^{t'-1} \alpha^{\tau} \|g^{\tau}\|^2 \geq \frac{\epsilon^2}{4} \sum_{\tau=t}^{t'-1} \alpha^{\tau} \geq \frac{\epsilon^2}{4} \frac{\epsilon(1-\zeta)}{4\lambda\beta}.$$

Since the number of such t is infinite, it follows that $\sum_{\tau=1}^{\infty} \alpha^{\tau} \|g^{\tau}\|^2 = \infty$, a contradiction of (29).

Since $\{g^t\} \rightarrow 0$ and (9) holds for all $s \in T$, it follows that $\{\nabla f(x_1^s)\}_{s \in T} \rightarrow 0$. $\square$

4. Convergence rate and stepsize analysis. In this section, we show that under a growth assumption on $\nabla f_1, \dots, \nabla f_m$ (see Assumption B below), the incremental gradient method either is linearly convergent in some sense or has its stepsize bounded away from zero (see Proposition 4.2). This result gives an explanation of the observed behavior that on some problems, the stepsize remains bounded away from zero (see the numerical experience reported in section 6).

To establish our result, we first need the following technical lemma.

LEMMA 4.1. *For any $\alpha^t \in (0, \alpha^0]$ and any $x_1^t, \dots, x_{m+1}^t$ in $\mathcal{R}_\rho^\eta$ satisfying (2), with $d_1^t, \dots, d_m^t$ given by (3), for $t = 0, 1, \dots$, we have*

$$(32) \quad \|d_i^t\| \leq \sum_{\tau=0}^{t-1} \mu_{t-\tau}^{t-\tau+1} h^\tau + (1 + \dots + \mu^{i-1}) h^t,$$

for $i = 1, \dots, m$ and $t = 0, 1, \dots$, where we let

$$(33) \quad h^t := \max_{i=1, \dots, m} \|\nabla f_i(x_1^t)\|, \quad t = 0, 1, \dots, \quad \mu := \left((m-1)\alpha^0 \max_{i=1, \dots, m} \lambda_i + \zeta \right)^{1/m}.$$

Proof. Clearly, (32) holds for $t = 0$ and $i = 1$. Suppose that, for some $s \geq 0$ and some $1 \leq j \leq m$, (32) holds for $i = 1, \dots, m$ if $t < s$ and for $i = 1, \dots, j$ if $t = s$. First, consider the case $j = m$. Then, by (3) with $t = s+1$ and (32) with $t = s$ and $i = m$,

$$\|d_1^{s+1}\| = \|\nabla f_1(x_1^{s+1}) + \zeta d_m^s\| \leq \|\nabla f_1(x_1^{s+1})\| + \zeta \|d_m^s\| \leq h^{s+1} + \zeta \left(\sum_{\tau=0}^s \mu_{s-\tau}^{s-\tau+1} h^\tau \right).$$

Since $\zeta \leq \mu^m$, this implies (32) holds for $t = s+1$ and $i = 1$. Second, consider the case $j < m$. Then, by (3) for $t = s$ and $i = j+1$,

$$\begin{aligned} \|d_{j+1}^s\| &= \|\nabla f_{j+1}(x_{j+1}^t) + \zeta d_j^t\| \\ &\leq \|\nabla f_{j+1}(x_{j+1}^t) - \nabla f_{j+1}(x_1^t)\| + \|\nabla f_{j+1}(x_1^t)\| + \zeta \|d_j^t\| \\ &\leq \alpha^t \lambda_{j+1} (\|d_j^t\| + \dots + \|d_1^t\|) + \|\nabla f_{j+1}(x_1^t)\| + \zeta \|d_j^t\| \\ &\leq j \alpha^t \lambda_{j+1} \left(\sum_{\tau=0}^{t-1} \mu_{t-\tau-1}^{t-\tau} h^\tau + (1 + \dots + \mu^{j-1}) h^t \right) \\ &\quad + h^t + \zeta \left(\sum_{\tau=0}^{t-1} \mu_{t-\tau-1}^{t-\tau} h^\tau + (1 + \dots + \mu^{j-1}) h^t \right) \\ &= (j \alpha^t \lambda_{j+1} + \zeta) \sum_{\tau=0}^{t-1} \mu_{t-\tau-1}^{t-\tau} h^\tau + (j \alpha^t \lambda_{j+1} + \zeta) (1 + \dots + \mu^{j-1}) h^t + h^t \\ &\leq \mu^m \sum_{\tau=0}^{t-1} \mu_{t-\tau-1}^{t-\tau} h^\tau + (1 + \dots + \mu^j) h^t, \end{aligned}$$

where the second inequality follows from $x_{j+1}^t, x_1^t \in \mathcal{R}_\rho^\eta$ and the Lipschitz continuity of ∇f_{j+1} on $\mathcal{R}_\rho^\eta$ (with constant λ_{j+1}) and (2), the third inequality follows from (32) with $i = 1, \dots, j$ and (33). This implies (32) holds for $t = s$ and $i = j+1$. Thus, by induction on t and i , (32) holds for all $i = 1, \dots, m$ and all $t = 0, 1, \dots$. $\square$

Consider the following growth assumption on $f_1, \dots, f_m$.

Assumption B. There exists $c_1 > 0$ such that

$$\max_{i=1, \dots, m} \|\nabla f_i(x)\| \leq c_1 \|\nabla f(x)\| \quad \forall x \in \mathcal{R}_\rho^\eta.$$

Assumption B roughly says that the size of $\nabla f_1, \dots, \nabla f_m$ should grow no faster than linearly with the size of ∇f . In particular, this assumption requires that $\nabla f_1(x) = \dots = \nabla f_m(x) = 0$ at any stationary point x of f . This requirement, though restrictive, is not entirely unrealistic for certain applications. For example, for the application of neural network training, this requirement amounts to being able to train the neural network to achieve zero output error on the learning examples. In fact, it is possible for this requirement to fail to hold and still have the stepsize bounded away from zero. Consider the example of $n = 1, m = 2$ and $f_1(x) = x, f_2(x) = -x$. Then, $f \equiv 0$ and Assumption A holds with $\eta = \rho = \infty$ and $\lambda_1 = \lambda_2 = 0$. Upon applying the incremental gradient method with, say, $\zeta = 0$ and any choice of x_1^0, ω, T , and $\epsilon_0, \epsilon_1, \epsilon_2, \epsilon_3$ satisfying $\epsilon_1 < 1$, we find that $\delta_2 = \delta_3 = g^t = p^t = 0$ and, in particular, $\alpha^t = \epsilon_0$ for all $t = 0, 1, \dots$. In contrast, the other stepsize rules mentioned in section 1 would require the stepsize to tend to zero on this example. (This example is degenerate in the sense that every $x \in \mathfrak{R}$ is a stationary point of f . However, it can be easily modified so that x_1^0 is not a stationary point, etc.) In general, if the iterates are in a region where $f_1, \dots, f_m$ are nearly linear, then the stepsize will tend not to decrease.

By using Lemma 4.1 and the fact (see the proof of Proposition 3.4) that (31) holds whenever $\alpha^s \neq \alpha^{s-1}$, we have the following convergence rate and stepsize result for the incremental gradient method. The result roughly says that, under Assumption B, either h^t given by (33) tends to zero linearly in some sense or α^t is bounded away from zero. The proof of this uses the idea that if h^t does not tend to zero linearly, then neither does $\|\nabla f(x_1^t)\|$ (by Assumption B) from which it can be shown that $\|d_i^t\| = O(\|\nabla f(x_1^t)\|)$ (see (35)). This in turn can be used to show that the right-hand side of (31) is bounded away from zero.

PROPOSITION 4.2. *Assume Assumption B (in addition to Assumption A) is satisfied and let $\{(x_1^t, \dots, x_{m+1}^t)\}_{t=0,1,\dots}$ and $\{\alpha^t\}_{t=0,1,\dots}$ be generated by the incremental gradient method (2)–(3) and (4)–(13) (which, by Prop. 3.4, are well defined) with α^0 chosen sufficiently small so that μ given by (33) is less than 1. If there exists a $c_2 \geq 1$ and $\sigma \in (\mu, 1)$ such that*

$$(34) \quad \sigma_{t-\tau}^{t-\tau+1} h^\tau \leq c_2 h^t, \quad \tau = 0, \dots, t-1,$$

for $t = 1, 2, \dots$, where h^t is given by (33), then $\liminf_{t \rightarrow \infty} \alpha^t > 0$.

Proof. Fix any $t \in \{1, 2, \dots\}$. Since (30) holds, then Lemma 4.1 and $c_2 \geq 1$ and $\sigma < 1$ show that, for each $i = 1, \dots, m$, d_i^t given by (3) satisfies

$$\begin{aligned} \|d_i^t\| &\leq \sum_{\tau=0}^{t-1} \mu_{t-\tau}^{t-\tau+1} h^\tau + \left(1 + \dots + \left(\frac{\mu}{\sigma}\right)^{m-1}\right) c_2 h^t \\ &\leq \sum_{\tau=0}^{t-1} \mu_{t-\tau}^{t-\tau+1} \left(\frac{c_2 h^t}{\sigma_{t-\tau}^{t-\tau+1}}\right) + \left(1 + \dots + \left(\frac{\mu}{\sigma}\right)^{m-1}\right) c_2 h^t \\ &\leq \sum_{\tau=0}^{t-1} \left(\frac{\mu}{\sigma}\right)_{t-\tau}^{t-\tau+1} c_2 h^t + \left(1 + \dots + \left(\frac{\mu}{\sigma}\right)^{m-1}\right) c_2 h^t \end{aligned}$$

$$\begin{aligned}
&= \sum_{\tau=0}^t \left(\frac{\mu}{\sigma}\right)_{t-\tau}^{t-\tau+1} c_2 h^t \\
&\leq \frac{c_2 h^t}{1 - \mu/\sigma} \\
(35) \quad &\leq c_3 \|\nabla f(x_1^t)\|,
\end{aligned}$$

where the second inequality follows from (34); the third inequality follows from the fact $(a_1 + \dots + a_m)/(b_1 + \dots + b_m) \leq \max_{i=1, \dots, m} a_i/b_i \leq (a_1/b_1 + \dots + a_m/b_m)$; the last inequality follows by using Assumption B and letting $c_3 := \frac{c_2}{1-\mu/\sigma} c_1$. Then (10) and (15) yield

$$\begin{aligned}
\|g^t - \nabla f(x_1^t)\| &\leq \sum_{i=1}^m \|\nabla f_i(x_i^t) - \nabla f_i(x_1^t)\| \\
&\leq \sum_{i=1}^m \lambda_i \|x_1^t - x_i^t\| \\
&= \sum_{i=1}^m \lambda_i \alpha^t \|d_1^t + \dots + d_{i-1}^t\| \\
(36) \quad &\leq \lambda \alpha^t m c_3 \|\nabla f(x_1^t)\|,
\end{aligned}$$

where d_i^t is given by (3), the second inequality follows from ∇f_i being Lipschitz continuous on $\mathcal{R}_\rho^\eta$ (with constant λ_i) and the equality follows from (2). Thus,

$$\|g^t\| \geq \|\nabla f(x_1^t)\| - \|g^t - \nabla f(x_1^t)\| \geq (1 - \lambda \alpha^t m c_3) \|\nabla f(x_1^t)\|$$

and, by (10) and (35)–(36),

$$\begin{aligned}
\beta^t &= \max \left\{ \|g^t\|, \sum_{i=1}^m \|d_i^t\| \right\} \\
&\leq \max \{ \|g^t - \nabla f(x_1^t)\| + \|\nabla f(x_1^t)\|, m c_3 \|\nabla f(x_1^t)\| \} \\
&\leq \max \{ \lambda \alpha^t m c_3 + 1, m c_3 \} \|\nabla f(x_1^t)\|.
\end{aligned}$$

Thus, if $\{\alpha^s\} \downarrow 0$, then (31) must hold for all s along some subsequence of T . On the other hand, the above two inequalities and $\|g^t\| \leq \beta$ (which hold for all t) show that the right-hand side of (31) is bounded away from zero, which implies α^s is bounded away from zero for all s along this subsequence, a contradiction to $\{\alpha^s\} \downarrow 0$. $\square$

Note that, for all t , $h^t \geq \frac{1}{n} \sum_{i=1}^m \|\nabla f_i(x_1)\| \geq \frac{1}{n} \|\nabla f(x_1^t)\|$ while, by $x_1^t \in \mathcal{R}_\rho^\eta$ and Assumption B, $h^t \leq c_1 \|\nabla f(x_1^t)\|$. Thus, under Assumption B, (34) is equivalent to

$$\sigma_{t-\tau}^{t-\tau+1} \|\nabla f(x_1^\tau)\| \leq c'_2 \|\nabla f(x_1^t)\|, \quad \tau = 0, 1, \dots, t-1,$$

for some constant c'_2 . In the very special case where $f_1 = \dots = f_m$ (so Assumption B holds trivially), the above proof can be modified to show that α^t generated by the incremental gradient method with $\zeta = 0$ (i.e., steepest descent) is always bounded away from zero.

5. Extension to constrained problems. In this section, we consider an extension of the incremental gradient method of section 2 to the problem of minimizing, over a nonempty closed convex set $\mathcal{X}$ of $\mathbb{R}^n$, the function f given by (1). Such constrained problems arise in neural network training where bounds are placed on the weights of the neural network and which corresponds to $\mathcal{X}$ being a box. Due to the presence of the constraint set $\mathcal{X}$, the formula for updating x_i^t and d_i^t need to be modified much as in [12]. We show that, analogous to Proposition 3.4, this extended method has desirable global convergence properties (see Proposition 5.4).

Consider the following iterative method for solving the preceding problem whereby, for a given $x_1^0 \in \mathcal{X}$, we generate a sequence $(x_1^t, \dots, x_{m+1}^t)$, $t = 0, 1, \dots$, according to

$$(37) \quad \begin{aligned} x_{i+1}^t &:= x_i^t - \alpha^t d_i^t, \quad i = 1, \dots, m, \\ x_1^{t+1} &:= [x_{m+1}^t]^+, \end{aligned}$$

where α^t is a positive scalar and

$$(38) \quad d_i^t := \begin{cases} \nabla f_i(x_i^t) + \zeta \nabla f_m(x_m^{t-1}) & \text{if } i = 1, \\ \nabla f_i(x_i^t) + \zeta \nabla f_{i-1}(x_{i-1}^t) & \text{if } i > 1 \end{cases}$$

with $x_m^{-1} = x_1^0$ given. (Here, $[\cdot]^+$ denotes the orthogonal projection operator onto $\mathcal{X}$. This operator can be evaluated fairly easily if $\mathcal{X}$ is either a box or a Euclidean sphere or simplex.) In the context of neural network training, the projection in (37) corresponds to the oft-used practice of truncating the weights of the neural network at their respective bounds. Note that (38) sets d_i^t to be a weighted sum of the two most recently computed gradients. This contrasts with (3), which sets d_i^t to be a weighted sum of *all* previously computed gradients. Thus, while (37) reduces to (2) when $\mathcal{X} = \mathbb{R}^n$, the formula (38) does not reduce to (3) when $\mathcal{X} = \mathbb{R}^n$, so (37)–(38) is a different method from (2)–(3). Our results can be extended to the case where d_i^t is a weighted sum of the K ($K \geq 1$ and constant) most recently computed gradients though, for simplicity, we will not consider this more general case here. It is an open question whether our results can be extended to the case where (38) is replaced by (3). Also, we note that in the case where $\zeta = 0$, the method (37)–(38) reduces to the approximate gradient-projection method studied in [12]. Our preliminary numerical experience suggests that $\zeta > 0$ (e.g., $\zeta = .1$) is typically preferable.

As with the incremental gradient method of section 2, we propose a new rule for choosing the stepsize α^t adaptively. To describe this new stepsize rule, we make the following assumption, analogous to Assumption A, about $f, f_1, \dots, f_m, \mathcal{X}$ and the initial iterate x_1^0 .

Assumption C. There exist scalars $\eta > f(x_1^0)$ and $\rho > 0$ such that, for $i = 1, \dots, m$, ∇f_i is bounded and Lipschitz continuous (with some constant λ_i) on the set

$$\mathcal{X}_\rho^\eta := \{ x \in \mathcal{X} : f(x) \leq \eta \} + \rho \mathcal{B},$$

where $\mathcal{B} = \{ x \in \mathbb{R}^n : \|x\| \leq 1 \}$.

Assumption C is quite mild and, in particular, is satisfied when $f_1, \dots, f_m$ are twice differentiable and the level set $\{ x \in \mathcal{X} : f(x) \leq \eta \}$ is bounded for some $\eta > f(x_1^0)$.

The new stepsize rule for choosing α^t , analogous to the one of section 2, depends on η, ρ and $\lambda_1, \dots, \lambda_m$ and periodically checks if a certain descent condition is satisfied since the previous check was made and, if not, decreases the stepsize and restarts

the method from when the previous check was made. Below, we formally state the method (37)–(38) using this stepsize rule. We will call this method the *incremental gradient-projection* method.

Incremental gradient-projection method (with 1-memory momentum term). Choose any $x_1^0 \in \mathcal{X}$ such that Assumption C holds for some η, ρ and $\lambda_1, \dots, \lambda_m$. Choose any $\zeta \in (0, \infty)$. Choose any $\omega \in (0, 1)$ and any subsequence T of $\{1, 2, \dots\}$ containing 1. Choose any positive scalars $\epsilon_0, \epsilon_1, \epsilon_2$ satisfying $\epsilon_1 < 1 + \zeta$.

Step 0. Let α^0 be the largest element of $\{\epsilon_0, \omega\epsilon_0, \omega^2\epsilon_0, \dots\}$ for which $x_2^0, \dots, x_m^0, x_1^1$ given by (37)–(38) with $t = 0$ satisfy the following two conditions:

$$(39) \quad (x_2^0, \dots, x_m^0, x_1^1) \in (\mathcal{X}_\rho^\eta)^m,$$

$$(40) \quad f(x_1^1) \leq \eta - \lambda_m \zeta (\alpha^0 \beta^0)^2,$$

where β^0 is given by (10).

Step 1. For each $s \in T$, let $h = \min\{t \in T : t > s\}$ and we generate $(x_1^h, d_m^{h-1}, \alpha^{h-1})$ from $(x_1^s, d_m^{s-1}, \alpha^{s-1})$ as follows: If $r^s = 0$, we stop. Else, we let α^{h-1} be the largest element of $\{\alpha^{s-1}, \omega\alpha^{s-1}, \omega^2\alpha^{s-1}, \dots\}$ for which $x_2^t, \dots, x_m^t, x_1^{t+1}$ given by (37)–(38) with $\alpha^t = \alpha^{h-1}$, $t = s, \dots, h-1$, satisfy the following three conditions:

$$(41) \quad (x_2^t, \dots, x_m^t, x_1^{t+1}) \in (\mathcal{X}_\rho^\eta)^m, \quad t = s, \dots, h-1,$$

$$(42) \quad f(x_1^h) \leq \eta - \epsilon_1 \sum_{\tau=1}^{h-1} \alpha^\tau \|\hat{r}^\tau\|^2 - \lambda_m \zeta (\alpha^{h-1} \beta^{h-1})^2,$$

$$(43) \quad \|r^s - \hat{r}^s\| \leq \epsilon_2 \|\hat{r}^s\|,$$

where, for $t = s, \dots, h-1$, we define

$$(44) \quad r^t := [x_1^t - \nabla f(x_1^t)]^+ - x_1^t, \quad \hat{r}^t := [x_1^t - g^t]^+ - x_1^t,$$

with g^t and β^t given by (10).

Like the stepsize rule of section 2, the above stepsize rule checks at each iteration $s \in T$ whether the current stepsize is acceptable (i.e., satisfies (41)–(43)) for all iterations between s and the next element h of T and, if not, it decreases the stepsize by ω and restarts the method from iteration s . Note that the conditions (40) and (42) are much simpler than their counterpart (6) and (8) of section 2. This is because here d_i^t depends only on the two most recently computed gradients. The quantity $\hat{r}^t$ may be viewed as an approximation to r^t , the “natural residual” at x_1^t .

To establish the global convergence of the incremental gradient-projection method, we first need the following three technical lemmas analogous to Lemmas 3.1–3.3. We assume throughout that Assumption C holds.

LEMMA 5.1. *For any $t \in \{0, 1, \dots\}$, any $\alpha^t > 0$, and any $x_1^t, \dots, x_m^t, x_1^{t+1}$ satisfying (37), we have, for $i = 1, \dots, m$, that*

$$\|\nabla f_i(x_1^t) - \nabla f_i(x_i^t)\| \leq \lambda_i \alpha^t \beta^t$$

whenever $x_1^t, x_i^t \in \mathcal{X}_\rho^\eta$ and that

$$\|\nabla f_i(x_1^{t+1}) - \nabla f_i(x_i^t)\| \leq 2\lambda_i \alpha^t \beta^t$$

whenever $x_i^t, x_1^{t+1} \in \mathcal{X}_\rho^\eta$, where β^t is given by (10).

Proof. We have from (37) and (10) that for any $i \in \{1, \dots, m+1\}$,

$$\|x_1^t - x_i^t\| = \alpha^t \left\| \sum_{l=1}^{i-1} d_l^t \right\| \leq \alpha^t \beta^t.$$

By a similar reasoning as above, we have $\|x_{m+1}^t - x_i^t\| \leq \alpha^t \beta^t$ which, together with the above inequality with $i = m+1$, yields

$$\begin{aligned} \|x_1^{t+1} - x_i^t\| &\leq \|x_1^{t+1} - x_{m+1}^t\| + \|x_{m+1}^t - x_i^t\| \\ &\leq \|x_1^t - x_{m+1}^t\| + \|x_{m+1}^t - x_i^t\| \\ &\leq \alpha^t \beta^t + \alpha^t \beta^t, \end{aligned}$$

where the second inequality follows from the observation that x_1^{t+1} is the point in $\mathcal{X}$ nearest in Euclidean distance to x_{m+1}^t and $x_1^t \in \mathcal{X}$. The above inequalities, together with ∇f_i being Lipschitz continuous (with constant λ_i) on $\mathcal{X}_\rho^\eta$ for $i = 1, \dots, m$, yield the desired results. $\square$

Under Assumption C, there exist positive scalars $\beta_1, \dots, \beta_m$ such that

$$(45) \quad \|\nabla f_i(x)\| \leq \beta_i \quad \forall x \in \mathcal{X}_\rho^\eta, \quad i = 1, \dots, m.$$

Let β and λ be given by (15). Also, let

$$(46) \quad \delta_0 := \rho / (2\beta(1 + \zeta)).$$

LEMMA 5.2. *For any $t \in \{0, 1, \dots\}$, any $\alpha^t \in (0, \delta_0]$, and any $x_m^{t-1}, x_1^t, \dots, x_m^t, x_1^{t+1}$ satisfying (37)–(38) and such that $x_m^{t-1} \in \mathcal{X}_\rho^\eta$, $x_1^t \in \mathcal{X}$ and $f(x_1^t) \leq \eta$, we have that $x_1^t, \dots, x_m^t$ are in $\mathcal{X}_\rho^\eta$, as is the line segment joining x_1^t with x_1^{t+1} .*

Proof. We claim that

$$(47) \quad \|x_l^t - x_1^t\| \leq \frac{\rho}{2\beta(1 + \zeta)} \sum_{k=1}^{l-1} (\beta_k + \zeta\beta_{k \ominus 1})$$

for $l = 1, \dots, m+1$. We prove this by induction on l . Clearly, (47) holds for $l = 1$. Suppose (47) holds for $l = 1, \dots, i$ for some $i \in \{1, \dots, m\}$ and we show below that it also holds for $l = i+1$. Since (47) holds for $l = 1, \dots, i$ and the right-hand side of (47) is bounded above by $\rho/2$, we have (cf. $f(x_1^t) \leq \eta$ and $x_1^t \in \mathcal{X}$) $x_1^t, \dots, x_i^t \in \mathcal{X}_\rho^\eta$, as well as $x_m^{t-1} \in \mathcal{X}_\rho^\eta$. Then, (37)–(38) and (45) and $\alpha^t \leq \delta_0$ yield

$$\begin{aligned} \|x_{i+1}^t - x_1^t\| &= \alpha^t \left\| \nabla f_i(x_i^t) + \zeta \nabla f_{i \ominus 1} \left(x_{i \ominus 1}^{\lfloor \frac{tm+i-2}{m} \rfloor} \right) \right\| \leq \alpha^t (\beta_i + \zeta\beta_{i \ominus 1}) \\ &\leq \frac{\rho}{2\beta(1 + \zeta)} (\beta_i + \zeta\beta_{i \ominus 1}). \end{aligned}$$

Since (47) holds for $l = i$, this shows that (47) holds for $l = i+1$.

The claim (47) together with $f(x_1^t) \leq \eta$ and $x_1^t \in \mathcal{X}$ implies that $x_l^t \in \mathcal{X}_\rho^\eta$ for $l = 1, \dots, m+1$. Also, since x_1^{t+1} is the point in $\mathcal{X}$ nearest in Euclidean distance to x_{m+1}^t (see (37)) and $x_1^t \in \mathcal{X}$, we have

$$\|x_{m+1}^t - x_1^{t+1}\| \leq \|x_{m+1}^t - x_1^t\| \leq \rho/2,$$

where the last inequality follows from (47) with $l = m + 1$ and then bounding the right-hand side of (47) from above by $\rho/2$. Thus,

$$\|x_1^t - x_1^{t+1}\| \leq \|x_1^t - x_{m+1}^t\| + \|x_{m+1}^t - x_1^{t+1}\| \leq \rho$$

and, hence, (cf. $f(x_1^t) \leq \eta$ and $x_1^t \in \mathcal{X}$) the line segment joining x_1^t with x_1^{t+1} lies in $\mathcal{X}_\rho^\eta$. $\square$

By using Lemma 5.1, we obtain the third lemma estimating the decrease in f -value per iteration of the incremental gradient-projection method.

LEMMA 5.3. *For any $t \in \{1, 2, \dots\}$, any $\alpha^\tau \in (0, 1/(1+\zeta)]$, and any $x_m^{\tau-1}, x_1^\tau, \dots, x_m^\tau, x_1^{\tau+1}$ satisfying (37)–(38) for $\tau = t-1, t$, and such that $x_1^t \in \mathcal{X}$ and both x_m^{t-1}, x_m^t and the line segment joining x_1^t with x_1^{t+1} lie in $\mathcal{X}_\rho^\eta$, we have*

$$(48) \quad f(x_1^{t+1}) \leq f(x_1^t) - (1 + \zeta)\alpha^t \|\hat{r}^t\|^2 + (1.5\lambda + 2\lambda_m\zeta)(\alpha^t\beta^t)^2 + \lambda_m\zeta(\alpha^{t-1}\beta^{t-1})^2,$$

where β^t is given by (10) and $\hat{r}^t$ is given by (44). Similarly, for any $\alpha^0 \in (0, 1/(1+\zeta)]$ and any $x_m^{-1}, x_1^0, \dots, x_m^0, x_1^1$ satisfying (37)–(38) (with $t = 0$) and such that $x_m^{-1} = x_1^0 \in \mathcal{X}$ and both x_m^0 and the line segment joining x_1^0 with x_1^1 lie in $\mathcal{X}_\rho^\eta$, we have

$$(49) \quad f(x_1^1) \leq f(x_1^0) - (1 + \zeta)\alpha^0 \|\hat{r}^0\|^2 + (1.5\lambda + \lambda_m\zeta)(\alpha^0\beta^0)^2.$$

Proof. Fix any $t \in \{1, 2, \dots\}$. Since the line segment joining x_1^t with x_1^{t+1} lies in $\mathcal{X}_\rho^\eta$ and, by $\nabla f = \nabla f_1 + \dots + \nabla f_m$ (cf. (1)) and Assumption C, ∇f is Lipschitz continuous (with constant λ given by (15)) on $\mathcal{X}_\rho^\eta$, we obtain from the intermediate value theorem (see [4, p. 639]) that

$$(50) \quad f(x_1^{t+1}) \leq f(x_1^t) + \langle \nabla f(x_1^t), x_1^{t+1} - x_1^t \rangle + .5\lambda \|x_1^{t+1} - x_1^t\|^2.$$

Using (37) and $x_1^t \in \mathcal{X}$, we bound the rightmost term in (50) as follows:

$$(51) \quad \|x_1^{t+1} - x_1^t\| = \left\| \left[x_1^t - \alpha^t \sum_{i=1}^m d_i^t \right]^+ - [x_1^t]^+ \right\| \leq \alpha^t \left\| \sum_{i=1}^m d_i^t \right\| \leq \alpha^t \beta^t,$$

where the first inequality follows from the nonexpansive property of the projection operator $[\cdot]^+$, and the last inequality follows from (10). The second term on the right-hand side of (50) can be bounded as follows:

$$\begin{aligned} & \langle \nabla f(x_1^t), x_1^{t+1} - x_1^t \rangle \\ &= \langle \nabla f(x_1^t) - g^t, x_1^{t+1} - x_1^t \rangle + \langle g^t, [x_1^t - \alpha^t(1 + \zeta)g^t]^+ - x_1^t \rangle \\ & \quad + \langle g^t, x_1^{t+1} - [x_1^t - \alpha^t(1 + \zeta)g^t]^+ \rangle \\ &\leq \|\nabla f(x_1^t) - g^t\| \|x_1^{t+1} - x_1^t\| - \frac{1}{\alpha^t(1 + \zeta)} \|[x_1^t - \alpha^t(1 + \zeta)g^t]^+ - x_1^t\|^2 \\ & \quad + \|g^t\| \|x_1^{t+1} - [x_1^t - \alpha^t(1 + \zeta)g^t]^+\| \\ &\leq \|\nabla f(x_1^t) - g^t\| \|x_1^{t+1} - x_1^t\| - \alpha^t(1 + \zeta) \|\hat{r}^t\|^2 + \|g^t\| \|x_1^{t+1} - [x_1^t - \alpha^t(1 + \zeta)g^t]^+\| \\ &\leq \|\nabla f(x_1^t) - g^t\| \|x_1^{t+1} - x_1^t\| - \alpha^t(1 + \zeta) \|\hat{r}^t\|^2 + \alpha^t \|g^t\| \left\| \sum_{i=1}^m d_i^t - (1 + \zeta)g^t \right\|, \end{aligned}$$

where the first inequality follows from the Cauchy–Schwarz inequality and the following well-known property of $[\cdot]^+$:

$$\langle x - y, [y]^+ - x \rangle \leq -\|[y]^+ - x\|^2 \quad \forall x \in \mathcal{X} \quad \forall y \in \mathbb{R}^n;$$

the second inequality follows from $\alpha^t(1+\zeta) \leq 1$ and $\|[x - \gamma d]^+ - x\| \geq \gamma\|[x - d]^+ - x\|$ for all $\gamma \in [0, 1]$ (see Lemma 1 in [7]) and (44); the last inequality follows from $x_1^{t+1} = [x_1^t - \alpha^t \sum_{i=1}^m d_i^t]^+$ (see (37)) and the nonexpansive property of $[\cdot]^+$. Also, we have from (10) and (38) that

$$\begin{aligned} \left\| \sum_{i=1}^m d_i^t - (1+\zeta)g^t \right\| &= \zeta \|\nabla f_m(x_m^{t-1}) - \nabla f_m(x_m^t)\| \\ &\leq \zeta (\|\nabla f_m(x_m^{t-1}) - \nabla f_m(x_1^t)\| + \|\nabla f_m(x_1^t) - \nabla f_m(x_m^t)\|) \\ &\leq \zeta \lambda_m (2\alpha^{t-1}\beta^{t-1} + \alpha^t\beta^t), \end{aligned}$$

where the last inequality follows from $x_m^{t-1}, x_1^t, x_m^t \in \mathcal{X}_\rho^\eta$ and Lemma 5.1. By a similar argument, we have that (24) holds. Combining the above two inequalities with (24) and $\|g^t\| \leq \beta^t$ (see (10)) yields

$$\begin{aligned} \langle \nabla f(x_1^t), x_1^{t+1} - x_1^t \rangle &\leq \lambda(\alpha^t\beta^t)^2 - \alpha^t(1+\zeta)\|\hat{r}^t\|^2 + \alpha^t\beta^t\zeta\lambda_m(2\alpha^{t-1}\beta^{t-1} + \alpha^t\beta^t) \\ &\leq \lambda(\alpha^t\beta^t)^2 - \alpha^t(1+\zeta)\|\hat{r}^t\|^2 + \zeta\lambda_m((\alpha^{t-1}\beta^{t-1})^2 + 2(\alpha^t\beta^t)^2), \end{aligned}$$

where the last inequality uses $ab \leq .5(a^2 + b^2)$. This together with (50)–(51) proves (48). The proof of (49) is very similar. $\square$

Below we state and prove the global convergence result for the incremental gradient-projection method. The proof uses Lemmas 5.1–5.3.

PROPOSITION 5.4. *The sequences $\{(x_1^t, \dots, x_{m+1}^t)\}_{t=0,1,\dots}$ and $\{\alpha^t\}_{t=0,1,\dots}$, generated by the incremental gradient-projection method (see (37)–(38) and (39)–(44)) are well defined. Moreover, either (i) $r^s = 0$ for some $s \in T$ or (ii) $\liminf_{t \rightarrow \infty} f(x_1^t) = -\infty$ or (iii) $\{\hat{r}^t\} \rightarrow 0$ and $\{r^t\}_{t \in T} \rightarrow 0$, where T is the subsequence of $\{1, 2, \dots\}$ specified in the method.*

Proof. We show by induction on t that α^t is well defined for $t = 0, 1, \dots$. Since $x_m^{-1} = x_1^0 \in \mathcal{X}$ and $f(x_1^0) < \eta$, for $\alpha^0 \leq \min\{\delta_0, 1/(1+\zeta)\}$, Lemma 5.2 shows that $x_1^0, \dots, x_m^0$ are in $\mathcal{X}_\rho^\eta$ as is the line segment joining x_1^0 with x_1^1 . Thus (39) holds and, by Lemma 5.3, (49) holds. Since $f(x_1^0) < \eta$, the latter implies that (40) holds for all α^0 sufficiently small, so α^0 , being the largest element of $\{\epsilon_0, \omega\epsilon_0, \omega^2\epsilon_0, \dots\}$ for which (40) holds, is well defined. Now assume that, for some $s \in T$, α^t is well defined for $t = 0, 1, \dots, s-1$. If $\nabla f(x_1^s) = 0$, then we stop with case (i). Suppose instead $\nabla f(x_1^s) \neq 0$ and let $h = \min\{t \in T : t > s\}$. We argue in the next two paragraphs that α^t is well defined for $t = s, \dots, h-1$.

Since $g^t \rightarrow \nabla f(x_1^s)$ and β^t is bounded as $\alpha^t \rightarrow 0$ for $t = s, \dots, h-1$, we have

(52)

$$\alpha^t \leq \min\{\delta_0, 1/(1+\zeta)\}, \quad \alpha^t \leq \frac{(1+\zeta - \epsilon_1)\|\hat{r}^t\|^2}{(1.5\lambda + 3\lambda_m\zeta)(\beta^t)^2}, \quad \alpha^s \leq \frac{\epsilon_2\|\hat{r}^s\|}{\lambda\beta^s}, \quad t = s, \dots, h-1,$$

for any $\alpha^s = \dots = \alpha^{h-1}$ sufficiently small. We claim that, whenever (52) holds, then

$$(53) \quad f(x_1^{t+1}) \leq \eta, \quad (x_2^t, \dots, x_m^t, x_1^{t+1}) \in (\mathcal{X}_\rho^\eta)^m,$$

for $t = s - 1, \dots, h - 1$. Clearly (53) holds for $t = s - 1$. (This follows from (39)–(40) when $s = 1$ and follows from (41) with $t = s - 1$ and (42) with $h = s$ when $s > 1$.) Suppose that, for some $k \in \{s, \dots, h - 1\}$, the relation (53) holds for $t = s - 1, \dots, k - 1$. From (37), we also have $x_1^{t+1} \in \mathcal{X}$ for $t = s - 1, \dots, k - 1$. Then, for each $t \in \{s, \dots, k\}$, we have $f(x_1^t) \leq \eta$, $x_m^{t-1} \in \mathcal{X}_\rho^\eta$ and $x_1^t \in \mathcal{X}$, which together with Lemma 5.2 and $\alpha^t \leq \delta_0$ (see (52)) imply that $x_1^t, \dots, x_m^t$ and the line segment joining x_1^t with x_1^{t+1} all lie in $\mathcal{X}_\rho^\eta$, so, by Lemma 5.3 and $\alpha^t \leq 1/(1 + \zeta)$ (see (52)), (48) holds. Then, using the second inequality in (52) to bound the right-hand side of (48) yields

$$(54) \quad f(x_1^{t+1}) \leq f(x_1^t) - \epsilon_1 \alpha^t \|\hat{r}^t\|^2 - \lambda_m \zeta (\alpha^t \beta^t)^2 + \lambda_m \zeta (\alpha^{t-1} \beta^{t-1})^2$$

for $t = s, \dots, k$. Also, we have that the inequality (42) holds for $h = s$. (For $s = 1$, this follows from (40); for $s > 1$, this follows from α^{s-1} being chosen in Step 1 such that (42) holds for $h = s$.) Summing (54) over all $t \in \{s, \dots, k\}$ and then adding to it this inequality, we obtain

$$(55) \quad f(x_1^{k+1}) \leq \eta - \epsilon_1 \sum_{t=1}^k \alpha^t \|\hat{r}^t\|^2 - \lambda_m \zeta (\alpha^k \beta^k)^2.$$

The right-hand side of this inequality is less than η , so the claim (53) holds for $t = k$. By induction on k , it follows that (53) holds for $t = s - 1, \dots, h - 1$.

Thus, if (52) holds, then (53) holds for $t = s - 1, \dots, h - 1$, so that (41) holds. In addition, our argument showed that (55) holds for $k = s, \dots, h - 1$, so that, upon letting $k = h - 1$ in (55), we obtain that (42) holds. To see that (43) holds, we use (44) and (24) with $t = s$ to obtain

$$\|r^s - \hat{r}^s\| = \|[x_1^s - \nabla f(x_1^s)]^+ - [x_1^s - g^s]^+\| \leq \|\nabla f(x_1^s) - g^s\| \leq \lambda \alpha^s \beta^s \leq \epsilon_2 \|\hat{r}^s\|,$$

where the first inequality uses the nonexpansive property of $[\cdot]^+$ and the last inequality is due to the last inequality in (52). Thus $\alpha^s, \dots, \alpha^{h-1}$, being the largest element of $\{\alpha^{s-1}, \omega \alpha^{s-1}, \dots\}$ such that (41)–(43) hold, are well defined. This completes the induction step and shows that α^t is well defined for all $t = 0, 1, \dots$

There are three cases: either (i) $r^s = 0$ for some $s \in T$ or else, since (42) holds for all $h \in T$, (ii) $\liminf_{t \rightarrow \infty} f(x_1^t) = -\infty$, or (iii)

$$(56) \quad \sum_{t=1}^{\infty} \alpha^t \|\hat{r}^t\|^2 < \infty.$$

Also, we have from $f(x_1^0) < \eta$ and (39) and (41) (with $h = \min\{t \in T : t > s\}$) for all $s \in T$ that

$$(57) \quad (x_1^t, \dots, x_m^t) \in (\mathcal{X}_\rho^\eta)^m, \quad t = 0, 1, \dots$$

We claim that, in case (iii), $\{\hat{r}^t\} \rightarrow 0$. Since $\{\alpha^t\}$ is monotonically decreasing, either $\liminf_{t \rightarrow \infty} \alpha^t > 0$ or $\{\alpha^t\} \downarrow 0$. In the first case, (56) yields $\{\hat{r}^t\} \rightarrow 0$. Consider now the second case. We showed earlier that, for each $s \in T$, (41)–(43) hold whenever (52) holds for $t = s, \dots, h - 1$, where $h = \min\{t \in T : t > s\}$. Then, the choice of $\alpha^s = \dots = \alpha^{h-1}$ being the largest element of $\{\alpha^{s-1}, \omega \alpha^{s-1}, \dots\}$ such that (41)–(43) hold implies either $\alpha^s = \alpha^{s-1}$ or

$$(58) \quad \frac{\alpha^s}{\omega} > \min_{t=s, \dots, h-1} \left\{ \min\{\delta_0, 1/(1 + \zeta)\}, \frac{(1 + \zeta - \epsilon_1) \|\hat{r}^t\|^2}{(1.5\lambda + 3\lambda_m \zeta)(\beta^t)^2}, \frac{\epsilon_2 \|\hat{r}^s\|}{\lambda \beta^s} \right\}.$$

Since (57) holds, then for $t = 0, 1, \dots$ we have from (44), (10), (45), and (15) that

$$\|\hat{r}^t\| = \|[x_1^t - g^t]^+ - [x_1^t]^+\| \leq \|g^t\| = \left\| \sum_{i=1}^m \nabla f_i(x_i^t) \right\| \leq \sum_{i=1}^m \beta_i = \beta$$

and from (38), (45), and (15) that

$$\sum_{i=1}^m \|d_i^t\| = \sum_{i=1}^m \left\| \nabla f_i(x_i^t) + \zeta \nabla f_{i \ominus 1} \left(x_{i \ominus 1}^{\lfloor \frac{tm+i-2}{m} \rfloor} \right) \right\| \leq \sum_{i=1}^m (\beta_i + \zeta \beta_{i \ominus 1}) = \beta(1 + \zeta),$$

so (10) yields $\beta^t \leq \beta(1 + \zeta)$. Since $\{\alpha^s\} \downarrow 0$ so that (58) holds for all s in some subsequence of T , it follows that $\hat{r}^t \rightarrow 0$ for t along some subsequence of $\{0, 1, \dots\}$. We now argue that $\hat{r}^t \rightarrow 0$ for t along the entire sequence $\{0, 1, \dots\}$. Suppose this is not the case so there exists $\epsilon > 0$ such that $\|\hat{r}^t\| > \epsilon$ for all t along some subsequence of $\{0, 1, \dots\}$. The following argument is a modification of the proof of [12, Prop. 2]. Consider any t such that $\|\hat{r}^t\| \geq \epsilon$. Since $\{\hat{r}^t\}_{t=0,1,\dots}$ contains a subsequence that tends to zero, there exists a smallest integer $t' > t$ such that $\|\hat{r}^{t'}\| < \epsilon/2$. Then (44) yields

$$\begin{aligned} \frac{\epsilon}{2} &\leq \|\hat{r}^t\| - \|\hat{r}^{t'}\| \\ &\leq \|\hat{r}^t - \hat{r}^{t'}\| \\ &= \left\| \left[x_1^t - \sum_{i=1}^m \nabla f_i(x_i^t) \right]^+ - x_1^t - \left[x_1^{t'} - \sum_{i=1}^m \nabla f_i(x_i^{t'}) \right]^+ + x_1^{t'} \right\| \\ &\leq \left\| \left(x_1^t - \sum_{i=1}^m \nabla f_i(x_i^t) \right) - \left(x_1^{t'} - \sum_{i=1}^m \nabla f_i(x_i^{t'}) \right) \right\| + \|x_1^t - x_1^{t'}\| \\ &\leq 2\|x_1^t - x_1^{t'}\| + \left\| \sum_{i=1}^m (\nabla f_i(x_i^t) - \nabla f_i(x_i^{t'})) \right\| \\ &= 2 \left\| \sum_{\tau=t}^{t'-1} \alpha^\tau \sum_{i=1}^m d_i^\tau \right\| + \left\| \sum_{\tau=t}^{t'-1} \sum_{i=1}^m (\nabla f_i(x_i^{\tau+1}) - \nabla f_i(x_1^{\tau+1}) + \nabla f_i(x_1^{\tau+1}) - \nabla f_i(x_i^\tau)) \right\| \\ &\leq 2 \sum_{\tau=t}^{t'-1} \alpha^\tau \beta^\tau + \sum_{\tau=t}^{t'-1} \sum_{i=1}^m \lambda_i (\alpha^{\tau+1} \beta^{\tau+1} + 2\alpha^\tau \beta^\tau) \\ &= \sum_{\tau=t}^{t'-1} (2\alpha^\tau \beta^\tau + \lambda(\alpha^{\tau+1} \beta^{\tau+1} + 2\alpha^\tau \beta^\tau)) \\ &\leq (2 + 3\lambda)\beta(1 + \zeta) \sum_{\tau=t}^{t'-1} \alpha^\tau, \end{aligned}$$

where the third inequality uses the nonexpansive property of $[\cdot]^+$; the second equality follows from (37); the fifth inequality follows from (10), (57), and Lemma 5.1; the last inequality follows from $\beta^\tau \leq \beta(1 + \zeta)$ for all τ and the monotone decreasing property of $\{\alpha^\tau\}_{\tau=0,1,\dots}$. We also have that $\|\hat{r}^\tau\| \geq \epsilon/2$ for $\tau = t, \dots, t' - 1$, which together with the above relation yields

$$\sum_{\tau=t}^{t'-1} \alpha^\tau \|\hat{r}^\tau\|^2 \geq \frac{\epsilon^2}{4} \sum_{\tau=t}^{t'-1} \alpha^\tau \geq \frac{\epsilon^2}{4} \frac{\epsilon}{2(2 + 3\lambda)\beta(1 + \zeta)}.$$

Since the number of such t is infinite, this implies $\sum_{\tau=1}^{\infty} \alpha^{\tau} \|\hat{r}^{\tau}\|^2 = \infty$, a contradiction of (56).

Since $\{\hat{r}^t\} \rightarrow 0$ and (43) holds for all $s \in T$, it follows that $\{r^s\}_{s \in T} \rightarrow 0$. $\square$

We may ask whether a convergence rate and stepsize result analogous to Proposition 4.2 holds for the incremental gradient-projection method. This, however, appears unlikely since the proof of Proposition 4.2 requires that $\nabla f_1(x) = \cdots = \nabla f_m(x) = 0$ at a stationary point x of the problem. Such an assumption is reasonable for an unconstrained problem but not for a constrained problem.

6. Implementation issues and numerical experience. To gain some insight into the implementation issues associated with the incremental gradient(-projection) method and its practical performance, we implemented the method to train a single-hidden-layer feedforward neural network, and compared the performance of the method (which, in this case, is effectively on-line backpropagation) with the conjugate gradient method using Polak–Ribiere update and Armijo stepsize rule. In this section we report our findings.

First, we briefly describe the problem of training a single-hidden-layer feedforward neural network (see [12, section 3] for a more detailed discussion). In this problem, we are given a collection of vectors $(I(1), O(1)), \dots, (I(m), O(m))$ in $\mathbb{R}^M \times \mathbb{R}^L$ (“training examples” of input and desired output), and the goal is to minimize the output error function f given by (1) with $f_i : \mathbb{R}^{MN+LN+N+L} \mapsto [0, \infty)$, $i = 1, \dots, m$, given by

$$f_i(u_1, \dots, u_N, v_1, \dots, v_N, \omega_1, \dots, \omega_N, z) = \left\| \sum_{k=1}^N v_k \sigma(\langle I(i), u_k \rangle + \omega_k) + z - O(i) \right\|^2,$$

where $u_1, \dots, u_N \in \mathbb{R}^M$, $v_1, \dots, v_N \in \mathbb{R}^L$, $\omega_1, \dots, \omega_N \in \mathbb{R}$, $z \in \mathbb{R}^L$, and with $\sigma : \mathbb{R} \mapsto \mathbb{R}$ (“sigmoidal activation function”) a user-chosen continuous function satisfying $\sigma(\theta) \rightarrow 1$ as $\theta \rightarrow \infty$ and $\sigma(\theta) \rightarrow 0$ as $\theta \rightarrow -\infty$. (In neural network terminology, N is the number of hidden neurons, $u_1, \dots, u_N, v_1, \dots, v_N$ are the weights on the neural connections, and $\omega_1, \dots, \omega_N, z$ are biases at the neurons.) In our testing we used the following standard choice of σ :

$$\sigma(\theta) = 1/(1 + \exp(-\theta/10))$$

and set N according to the specific training examples, as is done in practice. We have two specific test problems: For our first test problem, that of computing the parity of $\{0, 1\}$ -vectors [10, p. 131], we have $m = 5$, $M = 4$, $L = 1$, and

$$I(1) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, O(1) = 0, I(2) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, O(2) = 1, I(3) = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, O(3) = 0, \dots,$$

$$I(5) = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, O(5) = 0,$$

and we set $N = 1$. (Here the desired output is 0 when the input has an even number of 1's and, otherwise, the desired output is 1.) For our second test problem, that of

recognizing the characters of 0, 1, 2, 3, 4, we have $m = 5$, $M = 15$, $L = 3$, and

$$I(1) = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad O(1) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \dots, \quad I(5) = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \quad O(5) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix},$$

and we set $N = 3$. (Here the “1”s in $I(1)$, written as a 5×3 matrix, form the character 0 and similarly for $I(2), \dots, I(5)$. The desired output $O(i)$ is the binary representation of $i - 1$, for $i = 1, \dots, 5$.) We also experimented with a version of the problem in which the desired output was changed to $L = 1$ and $O(1) = 0, \dots, O(5) = 4$. However, though all methods converged faster on the problem, the trained neural network was much less accurate in recognizing corrupted input. Two key features of each function f_i are that, (i) it has multiple local minima, and (ii) evaluating ∇f_i requires roughly the same work as evaluating f_i .

Next, we describe the three methods we implemented. The first method, referred to as ALG1, is the incremental gradient method of section 2. For this method, we used the standard choice of $\epsilon_0 = 1$, $\omega = .5$, $T = \{1, 11, 21, 31, \dots\}$ and, after some experimentation, found the choice of $\zeta = .8$ to work best (much better than the memoryless choice of $\zeta = 0$). To avoid stepsizes becoming too small, it is desirable to choose η, ϵ_3 large and ϵ_1, ϵ_2 small and, for our implementation, we chose $\eta = 1.5f(x_1^0) + 100$, $\epsilon_3 = 1000$, $\epsilon_1 = \epsilon_2 = .00001$ and estimated $\lambda_1 + \dots + \lambda_m$ and ρ by 1 and ∞ , respectively. The second method, referred to as ALG2, is the incremental gradient-projection method of section 5 (with $\mathcal{X} = \mathbb{R}^n$). For this method, we used the standard choice of $\epsilon_0 = 1$, $\omega = .5$, $T = \{1, 11, 21, 31, \dots\}$ and, after some experimentation, found the choice of $\zeta = .1$ to work best. Analogous to ALG1, we chose $\eta = 1.5f(x_1^0) + 100$, $\epsilon_1 = .00001$, $\epsilon_2 = 1000$, and estimated λ_m and ρ by 1 and ∞ , respectively. The third method, referred to as ALG3, is the conjugate gradient method using the efficient Polak–Ribiere update and the Armijo stepsize rule [1, pp. 20, 57]. More precisely, the method generates, for any given $x^0 \in \mathbb{R}^n$, a sequence $x^0, x^1, \dots$ according to

$$x^{t+1} := x^t - \alpha^t d^t,$$

where

$$d^t := \begin{cases} \nabla f(x^t) + \beta^t d^{t-1} & \text{if } t > 0, \\ \nabla f(x^t) & \text{if } t = 0, \end{cases}$$

$$\beta^t := \langle \nabla f(x^t), \nabla f(x^t) - \nabla f(x^{t-1}) \rangle / \|\nabla f(x^{t-1})\|^2,$$

and α^t is the largest element of $\{\epsilon_0, \epsilon_0 \omega, \epsilon_0(\omega)^2, \dots\}$ for which x^{t+1} given above satisfies

$$f(x^{t+1}) \leq f(x^t) - \sigma \alpha^t \langle \nabla f(x^t), d^t \rangle.$$

Here $\epsilon_0 > 0, \omega \in (0, 1), \sigma \in (0, .5)$ are user-chosen parameters. In our implementation, we used the standard choice of $\epsilon_0 = 1, \omega = .5, \sigma = .1$, and, to ensure convergence, incorporated a steepest descent restart (i.e., replace d^t by $\nabla f(x^t)$ whenever $\langle \nabla f(x^t), d^t \rangle < .00001 \|\nabla f(x^t)\| \|d^t\|$). For all three methods, each component of the starting point (i.e., x_1^0 for ALG1, ALG2, and x^0 for ALG3) was randomly generated according to the uniform distribution on the interval $[0, 1]$ and the termination criterion was $f(x) \leq 10^{-8}$. All methods were coded in Matlab Version 4.2a and were run

TABLE 1
Performance of ALG1, ALG2, ALG3 on the two test problems.

Problem	ALG1		ALG2		ALG3	
	ngrad. ¹	nfunc ²	ngrad ¹	nfunc ²	ngrad ¹	nfunc ²
Parity	222.3	30.2	602.3	61.3	650.2	1775.0
Character Recognition	185.6	19.6	629.0	64.0	227.0	713.3

¹ ngrad denotes the number of times that $\nabla f_1, \dots, \nabla f_m$ have been evaluated.

² nfunc denotes the number of times that $f_1, \dots, f_m$ have been evaluated.

on a Decstation 5000. Table 1 gives the number of gradient and function evaluations for the three methods, averaged over three runs (all with a standard deviation of less than 20 percent).

From Table 1, it can be seen that ALG1 requires fewer gradient evaluations and function evaluations (which are the most expensive operations) than either ALG2 or ALG3. Since gradient evaluation requires roughly equal work as function evaluation so that the total work is roughly equal to the sum of gradient and function evaluations, we see that ALG1 requires less than one-third the total work of either ALG2 or ALG3, while ALG3 requires the most total work. Thus, for our test problems at least, we can draw the following conclusions: (i) the incremental gradient method using an unlimited-memory momentum term is more efficient than the conjugate gradient method using Polak–Ribiere update and Armijo stepsize rule; (ii) the incremental gradient method using an unlimited-memory momentum term is more efficient than that using a one-memory momentum term (which in turn is more efficient than that using no momentum term at all). We note that the stepsize rule is also crucial to the efficiency of the incremental gradient method. When we took ALG1 and replaced its stepsize rule by the well-known (nonadaptive) stepsize rule of

$$\alpha^t = c/t$$

(which produces stepsizes that are square summable but not summable), the convergence became agonizingly slow, regardless of the choice of the constant $c > 0$. In contrast, the stepsizes in both ALG1 and ALG2 remained at the value .5 after an initial decrease (the large stepsizes, as well as the presence of the momentum term, appear to be key to the good performance of ALG1), while the stepsizes in ALG3 varied between .125 and 1. On the other hand, we caution that these results are for some small test problems only and much more extensive testing is needed to determine the efficiency of the incremental gradient(-projection) method in general.

REFERENCES

- [1] D. P. BERTSEKAS, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.
- [2] D. P. BERTSEKAS, *Incremental least squares methods and the extended Kalman filter*, SIAM J. Optim., 6 (1996), pp. 807–822.
- [3] D. P. BERTSEKAS, *A new class of incremental gradient methods for least squares problems*, SIAM J. Optim., 7 (1997), pp. 913–926.
- [4] D. P. BERTSEKAS AND J. N. TSITSIKLIS, *Parallel and Distributed Computation: Numerical Methods*, Prentice–Hall, Englewood Cliffs, NJ, 1989.
- [5] W. C. DAVIDON, *New least-square algorithms*, J. Optim. Theory Appl., 18 (1976), pp. 187–197.
- [6] T. DENOEU AND R. LENGELLÉ, *Initializing back propagation networks with prototypes*, Neural Networks, 6 (1993), pp. 351–363.

- [7] E. M. GAFNI AND D. P. BERTSEKAS, *Two-metric projection methods for constrained optimization*, SIAM J. Control Optim., 22 (1984), pp. 936–964.
- [8] A. A. GAIVORONSKI, *Convergence properties of back-propagation for neural nets via theory of stochastic gradient methods, Part I*, Optim. Methods Software, 4 (1994), pp. 117–134.
- [9] L. GRIPPO, *A class of unconstrained minimization methods for neural network training*, Optim. Methods Software, 4 (1994), pp. 135–150.
- [10] J. HERTZ, A. KROGH, AND R. G. PALMER, *Introduction to the Theory of Neural Computation*, Addison–Wesley, Redwood City, CA, 1991.
- [11] Y. LE CUN, *Une procedure d'apprentissage pour reseau a seuil assymetrique*, in Proc. Cognitiva '85, Paris, France, pp. 599–604.
- [12] Z.Q. LUO AND P. TSENG, *Analysis of an approximate gradient projection method with applications to the backpropagation algorithm*, Optim. Methods Software, 4 (1994), pp. 85–101.
- [13] O. L. MANGASARIAN AND M. V. SOLODOV, *Serial and parallel backpropagation convergence via nonmonotone perturbed minimization*, Optim. Methods Software, 4 (1994), pp. 103–116.
- [14] M. F. MØLLER, *A scaled conjugate gradient algorithm for fast supervised learning*, Neural Networks, 6 (1993), pp. 525–533.
- [15] T. N. PAPPAS, *Solution of Nonlinear Equations by Davidon's Least Squares Method*, M.Sc. thesis, Department of Electrical Engineering, Massachusetts Institute of Technology, Cambridge, MA, 1982.
- [16] D. B. PARKER, *Learning-Logic*, Center for Computational Research in Economics and Management Science Report no. TR-47, Massachusetts Institute of Technology, Cambridge, MA, 1985.
- [17] B. T. POLYAK, *Introduction to Optimization*, Optimization Software, New York, 1987.
- [18] D. E. RUMELHART, G. E. HINTON, AND R. J. WILLIAMS, *Learning internal representations by error propagation*, in Parallel Distributed Processing—Explorations in the Microstructure of Cognition, Rumelhart and McClelland, eds., MIT Press, Cambridge, MA, 1986, pp. 318–362.
- [19] G. TESAURO, Y. HE, AND S. AHMAD, *Asymptotic convergence of back propagation*, Neural Comput., 1 (1989), pp. 382–391.
- [20] P. J. WERBOS, *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*, Ph.D. thesis, Committee on Applied Mathematics, Harvard University, Cambridge, MA, 1974.
- [21] P. J. WERBOS, *Backpropagation through time: What it does and how to do it*, in Proc. IEEE, 78 (1990), pp. 1550–1560.
- [22] H. WHITE, *Learning in artificial neural networks: A statistical perspective*, Neural Comput., 1 (1989), pp. 425–464.
- [23] H. WHITE, *Some asymptotic results for learning in single hidden-layer feedforward network models*, J. Amer. Statist. Assoc., 84 (1989), pp. 1003–1013.

AN EFFICIENT ALGORITHM FOR LARGE-SCALE NONLINEAR PROGRAMMING PROBLEMS WITH SIMPLE BOUNDS ON THE VARIABLES*

R. PYTLAK†

Abstract. In this paper we present a new conjugate gradient algorithm for nonlinear programming problems with simple bounds on the variables. The method is an extension of our conjugate gradient algorithm described in [R. Pytlak, *IMA J. Numer. Anal.*, 14 (1994), pp. 443–460]. The simple constraints on the variables are treated by a projection. Under mild assumptions the method is globally convergent. The algorithm has the property whereby many constraints can leave or enter an active set of constraints at one iteration. If the strict complementarity condition is satisfied, the active set at a solution is identified in a finite number of iterations. The algorithm has been tested on problems with more than 10,000 variables.

Key words. conjugate gradient algorithm, box constraints, projection, large-scale problems

AMS subject classifications. 65K05, 90C06, 90C30

PII. S1052623494276518

1. Introduction. In this paper, we consider an algorithm for the constrained minimization problem

$$(1.1) \quad \min_{x \in \mathcal{R}^n} f(x)$$

$$(1.2) \quad \text{subject to (s.t.) the simple bounds } l \leq x \leq u,$$

where we assume that l , u are fixed vectors and the inequalities are taken componentwise. In general, we assume that the function f is Lipschitz continuously differentiable (however, in some cases we will apply the stronger assumption that $f \in C^2$).

The method that we introduce in this paper is an extension of the conjugate gradient algorithm presented in [25] (some preliminary results for nondifferentiable problems were given in [24]).

If box constraints (1.2) are not present, then one can use a conjugate gradient algorithm in order to solve problem (1.1). The direction at the k th step of this algorithm is determined according to the rule

$$(1.3) \quad d_k = -\nabla f(x_k) + t_k d_{k-1},$$

where, e.g.,

$$(1.4) \quad t_k = \frac{\langle \nabla f(x_k) - \nabla f(x_{k-1}), \nabla f(x_k) \rangle}{\|\nabla f(x_{k-1})\|^2} \quad \text{or} \quad t_k = \frac{\|\nabla f(x_k)\|^2}{\|\nabla f(x_{k-1})\|^2},$$

see [2], [14], [15], [23]; more complicated formulae are also possible ([26], [27]). The first formula in (1.4) is usually called the Polak–Ribière formula while the second one is the Fletcher–Reeves formula. Usually the Polak–Ribière formula leads to algorithms with superior numerical properties.

*Received by the editors November 2, 1994; accepted for publication (in revised form) February 7, 1997.

<http://www.siam.org/journals/siopt/8-2/27651.html>

†Imperial College of Science, Technology, and Medicine, Centre for Process Systems Engineering, London, SW7 2BT, UK (ptk@ps.ic.ac.uk).

In the midseventies new conjugate gradient algorithms were constructed by Wolfe and Lemaréchal [28], [19] for solving nondifferentiable problems. Their bundle methods take a convex hull of the subgradients from current and previous iterations to define a direction of descent. If the function f is differentiable, then the Wolfe–Lemaréchal algorithm defines the search direction by

$$(1.5) \quad d_k = -\mathbf{Nr}\{\nabla f(x_k), -d_{k-1}\},$$

where $\mathbf{Nr}\{a, b\}$ is defined as the point from a line segment spanned by the vectors a and b which has the smallest norm, i.e.,

$$(1.6) \quad \|\mathbf{Nr}\{a, b\}\| = \min\{\|\lambda a + (1 - \lambda)b\| : 0 \leq \lambda \leq 1\}$$

and $\|\cdot\|$ is the Euclidean norm. Let us notice that the operation $\mathbf{Nr}\{\cdot, \cdot\}$ can be easily performed. This is a simple univariate quadratic problem with box constraints and can be solved analytically.

The rule (1.5) was mainly motivated by the need to have d_k as a direction of descent (in the nondifferentiable case, $\nabla f(x_k)$ is substituted by a bundle of subgradients).

Let us consider the problem

$$(1.7) \quad \begin{aligned} \min_{(\mu, d) \in \mathcal{R}^{n+1}} \quad & \{\mu + 1/2\|d\|^2\} \\ \text{s. t.} \quad & \langle \nabla f(x_k), d \rangle \leq \mu, \\ & -\langle d_{k-1}, d \rangle \leq \mu. \end{aligned}$$

We obtain the solution of this problem by solving its dual

$$\begin{aligned} \max_{0 \leq \lambda \leq 1} \quad & \{-1/2\|\lambda \nabla f(x_k) - (1 - \lambda)d_{k-1}\|^2\} \\ & = -1/2\|\mathbf{Nr}\{\nabla f(x_k), -d_{k-1}\}\|^2 \\ & = -1/2\|\lambda_k \nabla f(x_k) - (1 - \lambda_k)d_{k-1}\|^2. \end{aligned}$$

Moreover, the optimal value μ_k is

$$(1.8) \quad \mu_k = -\|d_k\|^2.$$

From this we can easily deduce the following properties:

$$(1.9) \quad \langle \nabla f(x_k), d_k \rangle \leq -\|d_k\|^2,$$

$$(1.10) \quad -\langle d_{k-1}, d_k \rangle \leq -\|d_k\|^2,$$

$$(1.11) \quad \langle \nabla f(x_k), d_k \rangle = -\|d_k\|^2, \quad \langle d_{k-1}, d_k \rangle = \|d_k\|^2, \quad \text{if } 0 < \lambda_k < 1,$$

because $(\lambda_k, 1 - \lambda_k)$ are the Lagrange multipliers for problem (1.7). From (1.9) we have that if $\|d_k\| \neq 0$, d_k is a direction of descent.

In [25] a new family of conjugate gradient algorithms has been introduced. The important difference between these methods and the Wolfe–Lemaréchal algorithm lies in a new direction finding subproblem

$$(1.12) \quad d_k = -\mathbf{Nr}\{\nabla f(x_k), -\beta_k d_{k-1}\}.$$

Let us notice that if $\beta_k = 1$, then we have the Wolfe–Lemaréchal algorithm. In [25] we showed that the Wolfe–Lemaréchal algorithm is, in fact, the Fletcher–Reeves

algorithm when directional minimization is exact. Moreover, we constructed the sequence $\{\beta_k\}$ such that directions generated by (1.12) are equivalent to those provided by the Polak–Ribière formula (under the assumption that directional minimization is exact). This sequence,

$$(1.13) \quad \beta_k = \frac{\|\nabla f(x_k)\|^2}{|\langle \nabla f(x_k) - \nabla f(x_{k-1}), \nabla f(x_k) \rangle|},$$

has striking resemblance to the Polak–Ribière formula and has not only superior numerical properties but also has good convergence properties better than those of all existing versions of the Polak–Ribière algorithm (see, e.g., [17], [27]).

To complete our introduction to algorithms for the unconstrained problem we should notice that the algorithm with the direction finding subproblem (1.12) has properties similar to (1.9)–(1.11),

$$(1.14) \quad \langle \nabla f(x_k), d_k \rangle \leq -\|d_k\|^2,$$

$$(1.15) \quad -\beta_k \langle d_{k-1}, d_k \rangle \leq -\|d_k\|^2,$$

$$(1.16) \quad \langle \nabla f(x_k), d_k \rangle = -\|d_k\|^2, \quad \beta_k \langle d_{k-1}, d_k \rangle = \|d_k\|^2, \quad \text{if } 0 < \lambda_k < 1.$$

If the simple constraints (1.2) are present in the formulation of the optimization problem, then the simple scheme of a conjugate gradient algorithm has to be modified. To cope efficiently with these constraints the projection algorithm is applied.

Let the projection operator $P[\cdot]$ be defined as follows:

$$(P[x])_i = \begin{cases} (l)_i & \text{if } (x)_i < (l)_i, \\ (x)_i & \text{if } (l)_i \leq (x)_i \leq (u)_i, \\ (u)_i & \text{if } (x)_i > (u)_i. \end{cases}$$

If d_k is a direction generated at the k th iteration of the projection algorithm, for example, $d_k = -\nabla f(x_k)$, then the next point is determined by finding α_k such that for $P[x_k + \alpha_k d_k]$

$$f(P[x_k + \alpha_k d_k]) - f(x_k) \leq \sigma(x_k) < 0,$$

where $\sigma(\cdot)$ is such that

$$\sigma(x) < 0, \quad \text{for all noncritical points } x,$$

and

$$\lim_{k \rightarrow \infty} \sigma(x_k) = 0 \quad \text{if } x_k \rightarrow x^*,$$

where x^* is a critical point of problem (1.1)–(1.2).

The projection algorithm was initially proposed by Levitin and Polyak [20] and Goldstein [18]. Then it was significantly improved by Bertsekas [1], who showed that a simple Armijo stepsize rule could be applied in it. Bertsekas' next important contribution was to show how scaling could be applied to a projection algorithm [3]. In particular, he constructed the globally convergent Newton's version of a projection algorithm. In the late eighties a projection method was further advanced by Conn, Gould, and Toint [10], [11] who extended a trust region algorithm to problems with

box constraints (1.2) using the projection operation. They also noted that it would be difficult to justify the Bertsekas algorithm [3] for nonconvex functions. Their results apply to nonconvex problems—their method has been tested on large-scale problems and is available in the LANCELOT package [12]. Recently, Byrd et al. [8] extended a variable storage quasi-Newton algorithm (L-BFGS algorithm described in [21]; see also [5], [16]) to problems with bounds on the variables. Numerical results of their algorithm, presented in [29], show that the method is efficient for large-scale problems. Our brief review of projection algorithms would not be complete without the excellent paper by Dunn [13], who gave a general approach to projection algorithms, and papers which discuss the important property of the projection algorithm of identifying the active set of constraints at a solution in a finite number of iterations (see, e.g., [6], [7]).

The organization of this paper is as follows. In section 2 we state our general algorithm together with its convergence properties. In section 3 we prove global convergence for one version of our algorithm. In section 4 we present numerical results of applying this version of our algorithm to large-scale test problems from the CUTE collection [4]. We also compare our method to the L-BFGS-B code described in [29].

Finally, we note that we are concerned with functions defined on the Euclidean space R^n ; $\|\cdot\|$ is the Euclidean norm, $\langle\cdot,\cdot\rangle$ is a scalar product. Moreover, by $\text{co}A$ we denote a convex hull of the set A . We denote by $(x)_i$ the i th component of a vector x . The list of frequently used symbols is presented at the end of the paper.

2. A general convergence theory. Before stating our algorithm we have to specify necessary optimality conditions which must be satisfied by any minimizer of problem (1.1)–(1.2).

LEMMA 2.1. *Let x^* be a feasible point. Then x^* is a critical point for problem (1.1)–(1.2) if and only if*

$$(2.1) \quad P[x^* - \alpha \nabla f(x^*)] = x^*$$

for all $\alpha > 0$. Moreover, condition (2.1) can be checked only for $\alpha = 1$.

Proof. The proof can be found in [1]. $\square$

In order to simplify the presentation we will consider the problem of minimizing f subject to the constraints

$$(2.2) \quad x \geq 0.$$

The more general case can be easily obtained by straightforward generalization of results for the problem with constraints (2.2).

The analogue of a conjugate gradient algorithm for problem (1.1)–(1.2) could take the form

$$x_{k+1} = P[x_k + \alpha_k d_k],$$

where d_k is given by (1.12), β_k is defined by (1.13), and $\alpha_k > 0$ is a solution to the directional minimization

$$(2.3) \quad \min_{\alpha > 0} f(P[x_k + \alpha d_k]).$$

Of course, the scheme (2.3) is impractical and the first thing in defining the algorithm is to introduce the proper directional minimization rule. Bertsekas used

the simple Armijo-like stepsize rule (for $d_k = -\nabla f(x_k)$)

$$(2.4) \quad \begin{aligned} & f(P[x_k - \beta^{m_k} \nabla f(x_k)]) - f(x_k) \leq \\ & -\mu \langle \nabla f(x_k), x_k - P[x_k - \beta^{m_k} \nabla f(x_k)] \rangle, \quad \mu \in (0, 1), \end{aligned}$$

where $\beta \in (0, 1)$ and m_k is the smallest integer number from the set $\{0, 1, \dots\}$ which guarantees (2.4).

In the case of our conjugate gradient algorithm, that rule could be stated as

$$f(P[x_k + \beta^{m_k} d_k]) - f(x_k) \leq -\mu \langle d_k, P[x_k + \beta^{m_k} d_k] - x_k \rangle.$$

Unfortunately, this simple scheme for the directional minimization would not result in an efficient algorithm because it is well known that some approximation of the exact minimization (2.3) is desired for the fast convergence of a conjugate gradient algorithm.

The function $f(P[x_k + \alpha d_k])$ can be interpreted as a composition of two functions: the first one is Lipschitzian and the second one continuously differentiable. Let us consider a piecewise linear *arc* of the form

$$(2.5) \quad x_k(\alpha) = x_k + d_k(\alpha), \quad \text{where } (d_k(\alpha))_i := \begin{cases} \alpha(d_k)_i & \text{if } \alpha \leq \alpha_k^i, \\ \alpha_k^i(d_k)_i & \text{if } \alpha > \alpha_k^i, \end{cases}$$

where the breakpoints $\{\alpha_k^i\}_1^n$ are calculated as follows:

$$(2.6) \quad \alpha_k^i := -\frac{(x_k)_i}{(d_k)_i}, \quad i = 1, \dots, n$$

(assuming that if $(d_k)_i \geq 0$ then $\alpha_k^i = \infty$). The points α_k^i can be ordered to define *bent* points of $x_k(\alpha)$: $0 \leq t_k^1 < t_k^2 < \dots < t_k^m$. One can notice that our function $P[x_k + \alpha d_k]$ is exactly of this form and that $f(P[x_k + \alpha d_k]) = f(x_k + d_k(\alpha))$ is differentiable at all $\alpha \geq 0$ but $\{t_k^i\}_1^m$. Let us denote by $f_{k,i}^\alpha(\beta)$ the function $\beta \rightarrow f((x_k(\alpha))_1, \dots, (x_k(\beta))_i, \dots, (x_k(\alpha))_n) / (d_k)_i$; then we can write

$$(2.7) \quad \nabla f_{k,i}^\alpha(\alpha) = \begin{cases} \nabla_{x_i} f(x_k + d_k(\alpha)) & \text{if } \alpha < \alpha_k^i, \\ 0 & \text{if } \alpha > \alpha_k^i, \end{cases} \quad i = 1, \dots, n.$$

At the breakpoint α_k^i , $f_{k,i}^{\alpha_k^i}(\cdot)$ is not differentiable and Clarke's generalized gradient [9] is expressed by the formula

$$\partial f_{k,i}^{\alpha_k^i}(\alpha_k^i) = \text{co} \{ \nabla_{x_i} f(x_k + d_k(\alpha_k^i)), 0 \}.$$

In this case we have freedom in choosing some vectors which could emulate the role of gradients in the directional minimization. Throughout the paper, we apply the following convention:

$$(2.8) \quad (g_k(\alpha))_i := \begin{cases} \nabla_{x_i} f(x_k + d_k(\alpha)) & \text{if } \alpha < \alpha_k^i, \\ 0 & \text{if } \alpha \geq \alpha_k^i, \end{cases} \quad i = 1, \dots, n.$$

Therefore, an ad hoc directional minimization could be as follows.

Rule 1. Find a positive number α_k such that

$$(2.9) \quad f(P[x_k + \alpha_k d_k]) - f(x_k) \leq -\mu \langle d_k, P[x_k + \alpha_k d_k] - x_k \rangle,$$

$$(2.10) \quad \langle g_k(\alpha_k), d_k \rangle \geq -\eta \|d_k\|^2, \quad 0 < \mu < \eta < 1.$$

Let us notice that if constraints (1.2) are not present then Rule 1 reduces to the rule that was applied in [25].

[25] Rule. Find a positive number α_k such that

$$\begin{aligned} f(x_k + \alpha_k d_k) - f(x_k) &\leq -\mu \alpha_k \|d_k\|^2, \\ \langle \nabla f(x_k + \alpha_k d_k), d_k \rangle &\geq -\eta \|d_k\|^2, \quad 0 < \mu < \eta < 1. \end{aligned}$$

Unfortunately, directional minimization rules (2.9)–(2.10), although they mimic the rules from nondifferentiable minimization, are still not satisfactory to guarantee convergence. The difficulty lies in the fact that the functional $f(P[x_k + \alpha d])$, which is nondifferentiable only at a finite number of points, is defined by $x_k(\alpha)$, which can have bent points close to zero. If such a situation happens in nondifferentiable optimization a new subgradient is calculated and added to the approximation of the generalized gradient. We do not know whether a similar scheme could be applied to modify rules (2.9)–(2.10). But even if it were possible we would have to use more than two vectors to define d_k .

To overcome the drawback of Rule 1 we make two modifications. First of all, if $(x_k)_i$ is close to 0 and $\nabla_{x_i} f(x_k) > 0$, then we set $(d_k)_i = -\nabla_{x_i} f(x_k)$. We know, for example from [1], that if $d_k = -\nabla f(x_k)$ and x_k is not a critical point then for sufficiently small values of α_k , condition (2.9) is satisfied. If $(x_k)_i$ is close to 0, $\nabla_{x_i} f(x_k) \leq 0$, and $(x_{k+1})_i = 0$, then we must have $(d_k)_i \leq 0$, which means that $\nabla_{x_i} f(x_k)(d_k)_i \geq 0$ does not contribute to the “descent property” of d_k . Thus we can expect that other components of d_k will guarantee sufficient decrease of f . This approach was exploited in [3]. That method of determining d_k has another advantage. Let us notice that if the strict complementarity condition is satisfied at a solution x^* , i.e., $\nabla_{x_i} f(x^*) > 0$ for all i such that $x_i^* = 0$, then these constraints are likely to be identified in a finite number of iterations if, instead of using $(d_k)_i$ (whose sign does not have to be related to the sign of $\nabla_{x_i} f(x_k)$), we use $-\nabla_{x_i} f(x_k)$.

Bearing in mind that we consider simpler constraints (2.2) we define the set of indices I_k^+

$$(2.11) \quad I_k^+ := \{i \in \overline{1, n} : (x_k)_i \leq \varepsilon_k \text{ and } \nabla_{x_i} f(x_k) > 0\},$$

where $\{\varepsilon_k\}$ is such that $\varepsilon_k > 0$ and

$$(2.12) \quad \lim_{k \in K} \|x_k - P[x_k - \nabla f(x_k)]\| = 0 \Leftrightarrow \lim_{k \in K} \varepsilon_k = 0$$

for any subsequence $\{x_k\}_{k \in K}$.

Let us notice that if $\{\varepsilon_k\}$ is calculated in the following way:

$$(2.13) \quad w_k = x_k - P[x_k - D \nabla f(x_k)], \quad \varepsilon_k = \min(\varepsilon, \|w_k\|), \quad \varepsilon > 0,$$

where D is a diagonal positive definite matrix, then condition (2.12) will be satisfied.

The sets I_k^+ are used to modify the direction finding subproblem. Instead of solving problem (1.12) we find a new direction according to the rule

$$(2.14) \quad d_k = -\mathbf{N} \mathbf{r} \{ \nabla f(x_k), -\beta_k d_{k-1}^+ \}.$$

Here d_{k-1}^+ is defined by

$$(2.15) \quad (d_{k-1}^+)_i := \begin{cases} (d_{k-1})_i & \text{if } i \notin I_k^+, \\ -\nabla_{x_i} f(x_k)/\beta_k & \text{if } i \in I_k^+. \end{cases}$$

Let us notice that according to (2.14) and (2.15) we have

$$(2.16) \quad (d_k)_i = -\nabla_{x_i} f(x_k) \quad \text{for } i \in I_k^+.$$

The second modification takes into account the fact that in order to find α_k we can minimize $f(P[x_k + \alpha_k d_k])$ on the subintervals $[t_k^l, t_k^{l+1})$, $l = 1, \dots, m-1$. We introduce, for a given α , the set of all indices such that $i \notin I_k^+$, or $i \in I_k^+$ and $\alpha_k^i > \alpha$:

$$(2.17) \quad I_k(\alpha) := \{i \in \overline{1, n} : i \notin I_k^+\} \cup \{i \in \overline{1, n} : i \in I_k^+ \text{ and } \alpha_k^i > \alpha\}.$$

Having defined $I_k(\alpha)$ we can partition d_k into two vectors in such a way that one of them, $d_k^-(\alpha)$, is as follows:

$$d_k^-(\alpha) := \{(d_k)_i\}_{i \in I_k(\alpha)}.$$

Eventually, we can state our new directional minimization rule.

Rule 2. Find a positive number α_k such that

$$(2.18) \quad \begin{aligned} f(P[x_k + \alpha_k d_k]) - f(x_k) &\leq -\mu \langle d_k, P[x_k + \alpha_k d_k] - x_k \rangle, \\ \langle g_k(\alpha_k), d_k \rangle &\geq -\eta \|d_k^-(\alpha_k)\|^2, \quad 0 < \mu < \eta < 1. \end{aligned}$$

Let us notice that condition (2.18) uses the vector $d_k^-(\alpha)$ instead of the vector that is defined by d_k and all indices which are not active at $x_k + d_k(\alpha)$ (in other words, we do not set to zero components of d_k which become active at α and which indices do not belong to I_k^+). This gives us a less tight directional minimization rule.

We can state our algorithm as follows.

ALGORITHM.

Parameters: $\mu, \eta \in (0, 1)$, $\eta > \mu$, $\varepsilon > 0$, $\{\beta_k\}$, diagonal positive definite matrix D

Data: x_0

1. Set $k = 0$,

$$\begin{aligned} d_{-1} &= -\nabla f(x_k), \\ \beta_k &= 1. \end{aligned}$$

2. Compute

$$(2.19) \quad w_k = x_k - P[x_k - D\nabla f(x_k)],$$

$$(2.20) \quad \varepsilon_k = \min(\varepsilon, \|w_k\|),$$

$$(2.21) \quad d_k = -\mathbf{Nr}\{\nabla f(x_k), -\beta_k d_{k-1}^+\}.$$

If $w_k = 0$ then STOP, otherwise

$$(2.22) \quad \text{set } (d_k)_i = 0 \text{ if } (x_k)_i = 0 \text{ and } (d_k)_i < 0.$$

3. Find a positive number α_k such that

$$(2.23) \quad f(P[x_k + \alpha_k d_k]) - f(x_k) \leq -\mu \langle d_k, P[x_k + \alpha_k d_k] - x_k \rangle,$$

$$(2.24) \quad \langle g_k(\alpha_k), d_k \rangle \geq -\eta \|d_k^-(\alpha_k)\|^2.$$

4. Substitute $P[x_k + \alpha_k d_k]$ for x_{k+1} , increase k by one, go to Step 2.

Remark 1. It is easy to show that all convergence results presented in this paper (except those stated in Theorem 3.2) are also valid if instead of using the scalar ε_k defined by (2.13) we apply

$$(2.25) \quad \varepsilon_k = \min(\varepsilon_1, \varepsilon_2 \|w_k\|),$$

where $\varepsilon_1, \varepsilon_2 > 0$. We can also use the scalar ε_k defined in the following way:

$$(2.26) \quad \varepsilon_k = \min(\varepsilon_1, \varepsilon_2 \|w_k\|_\infty),$$

where $\|\cdot\|_\infty$ is the l_∞ norm. If $\varepsilon_2 \geq 1$ then Theorem 3.2 is valid for both (2.25) and (2.26).

We can prove the following lemma.

LEMMA 2.2. *Assume that x_k is a noncritical point and $d_k \neq 0$ is calculated in Step 2 of the algorithm. Then there exists a positive α_k such that (2.23)–(2.24) are satisfied, or*

$$(2.27) \quad \lim_{\alpha \rightarrow \infty} f(P[x_k + \alpha d_k]) = -\infty.$$

Proof. Let us denote by $\tilde{d}_k$ the direction calculated in Step 2 before modifications (2.22) are made. From (1.14) we have

$$\langle \nabla f(x_k), \tilde{d}_k \rangle \leq -\|\tilde{d}_k\|^2.$$

Let $(x_k)_i = 0$. If $i \in I_k^+$, then

$$\nabla_{x_i} f(x_k)(\tilde{d}_k)_i = -(\tilde{d}_k)_i^2$$

by (2.16), and if $(\tilde{d}_k)_i$ is set to zero, then for the modified vector d_k we have

$$(2.28) \quad \langle \nabla f(x_k), d_k \rangle \leq -\|d_k\|^2.$$

If $i \notin I_k^+$, then $\nabla_{x_i} f(x_k) \leq 0$ by (2.11), so that $(\tilde{d}_k)_i < 0$ implies

$$\nabla_{x_i} f(x_k)(\tilde{d}_k)_i \geq 0$$

and the modified vector d_k will also satisfy (2.28). Therefore, d_k is a direction of descent.

It is evident that $t_k^1 > 0$ where t_k^1 is the first *bent* point along the direction d_k . It means that on the subinterval $[0, t_k^1)$ $f(P[x_k + \alpha d_k])$ is a differentiable function. Therefore, we can apply, for example, the algorithm described in [25] to find $\alpha_k \in [0, t_k^1)$ (if it exists) which satisfies (2.23)–(2.24).

If conditions (2.23)–(2.24) are not satisfied on the subinterval $[0, t_k^1)$ the directional minimization is continued on the next subinterval $[t_k^1, t_k^2)$ and only those variables vary which have not hit constraints. The directional minimization rules can be stated as follows:

$$f(P[x_k + \alpha d_k]) - f(x_k) \leq -\mu \left[\sum_{i \notin T_k^1} \alpha (d_k)_i^2 + \sum_{i \in T_k^1} t_k^1 (d_k)_i^2 \right],$$

$$\langle g_k(\alpha), d_k \rangle \geq -\eta \|d_k^-(\alpha)\|^2, \quad \alpha \in [t_k^1, t_k^2),$$

where T_k^1 is a set of indices of constraints which have become active at $\alpha = t_k^1$.

We also have

$$\sum_{i \notin T_k^1} \nabla_{x_i} f(x_k + t_k^1 d_k) (d_k)_i < -\eta \|d_k^-(t_k^1)\|^2,$$

because otherwise we would stop at t_k^1 . Then, either the point α_k lies in the current subinterval, or we move to the next subinterval $[t_k^2, t_k^3)$ and repeat the same considerations. If α_k is not located in any of the subintervals $[t_k^i, t_k^{i+1})$, $[t_k^m, \infty)$ then (2.27) holds. $\square$

A procedure which finds α_k satisfying (2.23)–(2.24) in a finite number of operations can be easily constructed following procedures described in [22], [25].

We recall that the procedure described in [25] generates a sequence of points $\{\alpha^l\}_0^\infty$ constructed in the following way.

If

$$f(P[x_k + \alpha^l d_k]) - f(x_k) \leq -(\mu + \varepsilon) \langle d_k, P[x_k + \alpha^l d_k] - x_k \rangle$$

($\eta > \mu + \varepsilon$, $\varepsilon > 0$) we set $\alpha_\mu^{l+1} = \alpha^l$, $\alpha_N^{l+1} = \alpha_N^l$ and $\alpha_\mu^{l+1} = \alpha_\mu^l$, $\alpha_N^{l+1} = \alpha^l$; otherwise, ($\alpha_\mu^0 = 0$, $\alpha_N^0 = \infty$). Next, we substitute α^{l+1} by $(\alpha_\mu^l + \alpha_N^l)/2$ if $\alpha_N^{l+1} < \infty$ or by $2\alpha^l$ if $\alpha_N^{l+1} = \infty$.

We can easily see from the proof of Lemma 1 in [25] that in order to prove our claim that this procedure finds α_k fulfilling (2.23)–(2.24) in a finite number of iterations we have to analyze carefully the case when

$$\lim_{l \rightarrow \infty} \alpha_\mu^l = \lim_{l \rightarrow \infty} \alpha_N^l = \hat{\alpha}$$

and $\hat{\alpha}$ is a *bent* point.

Therefore, let us assume that $\hat{\alpha}$ is a *bent* point. We have

$$(2.29) \quad f(P[x_k + \alpha_N^l d_k]) - f(x_k) > -(\mu + \varepsilon) \langle d_k, P[x_k + \alpha_N^l d_k] - x_k \rangle,$$

$$(2.30) \quad f(P[x_k + \alpha_\mu^l d_k]) - f(x_k) \leq -(\mu + \varepsilon) \langle d_k, P[x_k + \alpha_\mu^l d_k] - x_k \rangle.$$

If infinitely often

$$(2.31) \quad \langle g_k(\alpha_N^l), d_k \rangle \geq -\eta \|d_k^-(\alpha_N^l)\|^2$$

then, because $f(P[x_k + \alpha d_k])$ is continuous and $\varepsilon > 0$, after a finite number of iterations both (2.23) and (2.24) will be satisfied.

To prove (2.31) we consider the inequality

$$(2.32) \quad f(P[x_k + \hat{\alpha} d_k]) - f(x_k) \leq -(\mu + \varepsilon) \langle d_k, P[x_k + \hat{\alpha} d_k] - x_k \rangle,$$

which follows from (2.30). Equation (2.32) together with (2.29) imply that

$$(2.33) \quad \begin{aligned} f(P[x_k + \alpha_N^l d_k]) - f(P[x_k + \hat{\alpha} d_k]) &> -(\mu + \varepsilon) \langle d_k, P[x_k + \alpha_N^l d_k] \\ &\quad - P[x_k + \hat{\alpha} d_k] \rangle \\ &= -(\mu + \varepsilon) (\alpha_N^l - \hat{\alpha}) \|\hat{d}_k\|^2 \end{aligned}$$

for sufficiently large l . Here $\hat{d}_k$ is the vector d_k modified in such a way that if $(x_k(\hat{\alpha}))_i = 0$ (see (2.5)) then $(\hat{d}_k)_i = 0$. On the other hand, because f is continuously differentiable on the right of $\hat{\alpha}$ (for sufficiently large l), we have

$$f(P[x_k + \alpha_N^l d_k]) - f(P[x_k + \hat{\alpha} d_k]) = (\alpha_N^l - \hat{\alpha}) \langle g_k(\alpha_N^l), d_k \rangle + o(\alpha_N^l - \hat{\alpha}).$$

(Here, $o(\alpha)$ is a higher-order term with respect to α .) Thus, from (2.33),

$$\langle g_k(\alpha_N^l), d_k \rangle + \frac{o(\alpha_N^l - \hat{\alpha})}{\alpha_N^l - \hat{\alpha}} > -(\mu + \varepsilon) \|\hat{d}_k\|^2 > -\eta \|d_k^-(\alpha_N^l)\|^2,$$

for sufficiently large l , where the last inequality follows from the fact that $\|\hat{d}_k\| \leq \|d_k^-(\alpha)\|$ due to the definition of $\hat{d}_k$ and $d_k^-(\alpha)$. Therefore, (2.31) must hold.

To investigate the convergence of the algorithm we begin by providing a crucial lemma which requires the following assumption.

ASSUMPTION 1. *There exists $L < \infty$ such that*

$$\|\nabla f(y) - \nabla f(x)\| \leq L\|y - x\|$$

for all x, y from a bounded set.

LEMMA 2.3. *If Assumption 1 holds, the direction d_k is determined by (2.19)–(2.22) and the stepsize coefficient α_k satisfies (2.23)–(2.24), then for any bounded subsequence $\{x_k\}_{k \in K}$*

$$(2.34) \quad \lim_{k \in K} \|x_k - P[x_k + d_k]\| = 0.$$

Proof. Because $\{f(x_k)\}_{k \in K}$ is nonincreasing and bounded from below (since $\{x_k\}_{k \in K}$ is bounded), it has to be convergent. From (2.23) and the theorem on three sequences we have

$$(2.35) \quad \begin{aligned} \lim_{k \in K} [f(x_{k+1}) - f(x_k)] &= 0, \\ \lim_{k \in K} \mu \langle d_k, P[x_k + \alpha_k d_k] - x_k \rangle &= 0. \end{aligned}$$

Because $\{d_k\}_{k \in K}$ is bounded, namely,

$$\|d_k\| \leq \|\nabla f(x_k)\| \quad \forall k \in K$$

by (2.14), the sequence $\{x_k - P[x_k + d_k]\}_{k \in K}$ is also bounded. Assume that (2.34) does not hold. It means that there exists a subsequence of $\{x_k - P[x_k + d_k]\}_{k \in K}$ which is convergent and whose limit does not satisfy (2.34). Without loss of generality we can assume that the sequence $\{x_k - P[x_k + d_k]\}_{k \in K}$ itself does not converge to zero. Furthermore, because $\{x_k\}_{k \in K}$, $\{d_k\}_{k \in K}$ are bounded, we can also extract their convergent subsequences. We also assume the following (again without loss of generality):

$$\begin{aligned} \lim_{k \in K} x_k &= \bar{x}, \\ \lim_{k \in K} d_k &= \bar{d}. \end{aligned}$$

Thus a contradiction to (2.34) implies that there exists at least one j such that

$$(2.36) \quad (\bar{x})_j > 0 \quad \text{and} \quad (\bar{d})_j \neq 0,$$

or

$$(2.37) \quad (\bar{x})_j = 0 \quad \text{and} \quad (\bar{d})_j > 0.$$

For any index i we have

$$(2.38) \quad (d_k)_i(P[x_k + \alpha_k d_k] - x_k)_i = \begin{cases} \alpha_k (d_k)_i^2 & \text{if } \alpha_k < \alpha_k^i, \\ -(x_k)_i (d_k)_i & \text{if } \alpha_k \geq \alpha_k^i, \end{cases} \geq 0,$$

since $(d_k)_i \leq 0$ if $\alpha_k \geq \alpha_k^i$.

Therefore, from (2.35), for each $i \in \overline{1, n}$ we have

$$(2.39) \quad \lim_{k \in K} (d_k)_i (P[x_k + \alpha_k d_k] - x_k)_i = 0.$$

Let j satisfy (2.36)–(2.37). If $\alpha_k < \alpha_k^j$ for infinitely many $k \in K_1 \subset K$ then, because $(d_k)_j \neq 0$ for sufficiently large $k \in K_1$ by (2.36)–(2.37),

$$\lim_{k \in K_1} \alpha_k = 0$$

from the first part of (2.38).

If $\alpha_k^j \leq \alpha_k$ for infinitely many $k \in K_2 \subset K$ then

$$\lim_{k \in K_2} (x_k)_j = 0$$

from the second part of (2.38), since $(d_k)_j \neq 0$ for sufficiently large $k \in K_2$ by (2.36)–(2.37). Therefore, by (2.37), $(d_k)_j > 0$ for sufficiently large $k \in K_2$. However, the above and (2.6), together with (2.22), imply that $\alpha_k^j = -\infty$ for sufficiently large $k \in K_2$ which contradicts the definition of K_2 . This guarantees that there exists an index $\bar{k} \in K$, such that $k \in K$ and $k \geq \bar{k}$ imply $k \in K_1$. Thus,

$$(2.40) \quad \lim_{k \in K} \alpha_k = 0.$$

Now we will consider separately two cases:

Case 1.

$$(2.41) \quad \lim_{k \in K} \|x_k - P[x_k - \nabla f(x_k)]\| \neq 0.$$

Case 2.

$$(2.42) \quad \lim_{k \in K} \|x_k - P[x_k - \nabla f(x_k)]\| = 0.$$

Moreover, the analysis will concern only those indices $i \in \overline{1, n}$ for which

$$(2.43) \quad \lim_{k \in K} (d_k)_i = (\bar{d})_i \neq 0.$$

As we will show later this assumption does not invalidate the proof.

Case 1. First we show that

$$(2.44) \quad \nabla_{x_i} f(x_k)(d_k)_i \geq 0 \quad \forall i \notin I_k^+ \text{ such that } \alpha_k^i \leq \alpha_k.$$

In order to prove (2.44), due to (2.11), we have to show that $(x_k)_j \leq \varepsilon_k$ for all j such that $j \notin I_k^+$, $\alpha_k^j \leq \alpha_k$ since then $\nabla_{x_j} f(x_k) \leq 0$ and $(d_k)_j \leq 0$. Because (2.41) holds we also have

$$(2.45) \quad \lim_{k \in K} \min(\varepsilon, \|x_k - DP[x_k - \nabla f(x_k)]\|) = \lim_{k \in K} \varepsilon_k = \bar{\varepsilon} > 0,$$

and due to our restriction (2.43), from (2.39) we have

$$(2.46) \quad \lim_{k \in K} (x_k)_j = 0$$

for all j such that $\alpha_k^j \leq \alpha_k$. Equations (2.45) and (2.46) imply that $(x_k)_j \leq \varepsilon_k$ for all these j which proves (2.44).

α_k has to satisfy the relation

$$\begin{aligned} \langle g_k(\alpha_k), d_k \rangle &= \sum_{i: \alpha_k^i > \alpha_k} \nabla_{x_i} f(x_k + d_k(\alpha_k))(d_k)_i \\ &\geq -\eta \|d_k^-(\alpha_k)\|^2. \end{aligned}$$

From Assumption 1 we have

$$\|d_k\| \|d_k(\alpha_k)\| L + \sum_{i: \alpha_k^i > \alpha_k} \nabla_{x_i} f(x_k)(d_k)_i \geq -\eta \|d_k^-(\alpha_k)\|^2,$$

and, from (2.17) and (2.44), we can write

$$(2.47) \quad \|d_k(\alpha_k)\| \|d_k\| L + \sum_{i \in I_k(\alpha_k)} \nabla_{x_i} f(x_k)(d_k)_i \geq -\eta \|d_k^-(\alpha_k)\|^2.$$

We can easily show that

$$\|d_k(\alpha_k)\| \leq \alpha_k \|d_k\|,$$

together with

$$\sum_{i \in I_k(\alpha_k)} \nabla_{x_i} f(x_k)(d_k)_i \leq -\|d_k^-(\alpha_k)\|^2$$

(which follows from (2.17), the definition of $d_k^-(\alpha_k)$, and (1.14)), and (2.47) imply that

$$(2.48) \quad (1 - \eta) \|d_k^-(\alpha_k)\|^2 \leq \alpha_k \|d_k\|^2 L.$$

Because of (2.40), in order to prove the lemma we have to show that

$$(2.49) \quad \lim_{k \in K} \|d_k^-(\alpha_k)\| \neq 0.$$

Indeed, if (2.49) does not hold then there exists at least one j which satisfies (2.36)–(2.37) and does not belong to $I_k(\alpha_k)$ for sufficiently large $k \in K$. This means that $j \in I_k^+$ and $\alpha_k^j \leq \alpha_k$. It follows, from (2.38), (2.39), that $\lim_{k \in K} (x_k)_j = 0$ which contradicts (2.37) because $j \in I_k^+$ and we must have $(d_k)_j = -\nabla_{x_j} f(x_k) < 0$ for sufficiently large $k \in K$.

Case 2. Contrary to Case 1, we cannot guarantee that $(x_k)_j \leq \varepsilon_k$ for all j such that $j \notin I_k^+$, $\alpha_k^j \leq \alpha_k$ and thus that (2.44) holds.

If we assume that (2.42) is true and $\lim_{k \in K} (x_k)_i = (\bar{x})_i > 0$ then $\nabla_{x_i} f(\bar{x}) = 0$. If $\lim_{k \in K} (x_k)_i = (\bar{x})_i = 0$ and $\nabla_{x_i} f(x_k)(d_k)_i < 0$ for infinitely many $k \in K$, then from (2.37) and (2.42) (which implies $\nabla_{x_i} f(\bar{x}) \geq 0$) we also have $\nabla_{x_i} f(\bar{x}) = 0$.

This means that instead of (2.47) we can use the inequality

$$\|d_k(\alpha_k)\| \|d_k\| L + \sum_{i \in I_k(\alpha_k)} \nabla_{x_i} f(x_k)(d_k)_i - P_k \geq -\eta \|d_k^-(\alpha_k)\|^2,$$

where $\{P_k\}$ is such that $\lim_{k \in K} P_k = 0$.

For large $k \in K$ we come to the contradiction

$$(2.50) \quad (1 - \eta) \|d_k^-(\alpha_k)\|^2 \leq \alpha_k \|d_k\|^2 L - P_k.$$

To complete the proof notice that (2.43) was needed to show that (2.44) is valid in Case 1). If (2.43) is not satisfied for some i then instead of (2.48) we will obtain (2.50) where $P_k \rightarrow_{k \in K} 0$. $\square$

For the convenience of future notations we assume that variables $(x)_i$ have been reordered in such a way that d_k can be partitioned into two vectors (d_k^1, d_k^2) where the first vector d_k^1 is represented by

$$d_k^1 := \{(d_k)_i\}_{i \notin I_k^+}.$$

The same convention applies to other vectors, for example,

$$(2.51) \quad \begin{aligned} \nabla f(x_k) &= ((\nabla f(x_k))^1, (\nabla f(x_k))^2) := (\nabla^1 f(x_k), \nabla^2 f(x_k)), \\ d_{k-1}^+ &= ((d_{k-1}^+)^1, (d_{k-1}^+)^2) = (d_{k-1}^1, (d_{k-1}^+)^2). \end{aligned}$$

It is important to remember that this *partitioning is always related to I_k^+* and not to I_{k-1}^+ as the second part of (2.51) would suggest. This partitioning simplifies statements and proofs of several convergence results presented later in the paper. It also enables the adaptation of the convergence analysis stated in [25] to the new algorithm.

The condition (2.34) stated in Lemma 2.3 is not, unfortunately, equivalent to the condition

$$\lim_{k \in K} \|x_k - P[x_k - \nabla f(x_k)]\| = 0,$$

which we have to prove. This is due to the additional vector $\beta_k d_{k-1}$ in the formula (2.14) which can also be restated, for components $i \notin I_k^+$, as

$$d_k^1 = -(1 - \lambda_k) \nabla^1 f(x_k) + \lambda_k \beta_k d_{k-1}^1,$$

where $0 \leq \lambda_k \leq 1$.

It can happen that $\lim_{k \rightarrow \infty} \|d_k^1\| = 0$ because the vectors d_{k-1}^1 are not appropriately scaled by β_k .

Moreover, we may have $\lim_{k \in K} \text{rd}(-\nabla^1 f(x_k), d_{k-1}^1) = \pi$ for certain sequences $\{\nabla f(x_k)\}_{k \in K}$ and an inexact line search. Here, $\text{rd}(a, b)$ is the angle between vectors a and b .

In each of these situations we shall have (2.34). Thus in order to prove the convergence of the algorithm we have to exclude these situations.

THEOREM 2.4. *Suppose that Assumption 1 is satisfied. Moreover, assume that for any convergent subsequence $\{x_k\}_{k \in K}$ whose limit is not a critical point,*

(i)

$$(2.52) \quad \liminf_{k \in K} (\beta_k \|d_{k-1}^1\|) \geq \nu_1 \liminf_{k \in K} \|\nabla^1 f(x_k)\|$$

where ν_1 is some positive constant;

(ii) there exists a number ν_2 such that $\nu_2 \in (0, 1)$,

$$(2.53) \quad \langle \nabla^1 f(x_k), d_{k-1}^1 \rangle \leq \nu_2 \|\nabla^1 f(x_k)\| \|d_{k-1}^1\|, \text{ whenever } \lambda_k \in (0, 1).$$

Then $\lim_{k \rightarrow \infty} f(x_k) = -\infty$, or every accumulation point of the sequence $\{x_k\}_0^\infty$ generated by the algorithm is a critical point.

Proof. Let us assume that

$$(2.54) \quad \bar{x} \neq P[\bar{x} - \nabla f(\bar{x})],$$

where $\bar{x}$ is the accumulation point of $\{x_k\}_{k \in K}$. If (2.54) holds then two cases can occur:

- 1) $(\bar{x})_i = 0$ and $\nabla_{x_i} f(\bar{x}) < 0$ for some $i \in \overline{1, n}$,
- 2) $(\bar{x})_i \neq 0$ and $\nabla_{x_i} f(\bar{x}) \neq 0$ for some $i \in \overline{1, n}$.

If case 1) occurs then there exists k_1 such that $i \notin I_k^+$ for $k \geq k_1$.

In the second case, if there exists k_2 such that for $k \geq k_2$, $i \in I_k^+$, then $\nabla_{x_i} f(x_k) > 0$, $(d_k)_i = -\nabla_{x_i} f(x_k)$, and from Lemma 2.3 we get $\nabla_{x_i} f(\bar{x}) = 0$, which is a contradiction.

Therefore, if (2.54) is satisfied then there exists k_3 such that for all $k \geq k_3$ there exists $i \notin I_k^+$ for which either 1) or 2) holds. This means that

$$\liminf_{k \in K} \|\nabla^1 f(x_k)\| > 0.$$

Now we will show that

$$\lim_{k \in K} \|d_k^1\| = 0.$$

We have

$$(2.55) \quad (P[x_k + d_k])_i - (x_k)_i = \begin{cases} (d_k)_i, \\ -(x_k)_i \text{ can occur if } (d_k)_i < 0. \end{cases}$$

Therefore, from Lemma 2.3, either

$$(2.56) \quad \lim_{k \in K} (d_k)_i = 0,$$

or

$$(2.57) \quad \lim_{k \in K} (x_k)_i = 0.$$

Let us denote by I^0 all indices for which (2.56) does not hold and which do not belong to I_k^+ . From (2.57) and the fact that

$$(2.58) \quad \lim_{k \in K} \varepsilon_k \geq \bar{\varepsilon} > 0$$

(because $\bar{x}$ is not a critical point) we have $\nabla_{x_i} f(x_k) \leq 0 \forall i \in I^0$ and, from the second case in (2.55), $(d_k)_i < 0 \forall i \in I^0$. Therefore,

$$0 \leq \sum_{i \in I^0} \nabla_{x_i} f(x_k) (d_k)_i \leq -\|d_k\|^2 - \sum_{i \notin I^0} \nabla_{x_i} f(x_k) (d_k)_i < 0$$

for sufficiently large $k \in K$. This is not possible; thus (2.57) cannot happen.

The rest of the proof can be carried out in a similar way as the proof of cases a, b, and c in the proof of Theorem 1 in [25] and therefore it is omitted. $\square$

The condition (2.53) is independent of the choice of the sequence $\{\beta_k\}$ and is related to the directional minimization. Therefore, it is not surprising that in some important situations it can be substituted by another, more easily verifiable condition.

LEMMA 2.5. *Suppose that Assumption 1 is satisfied. Assume that $\{x_k\}$ is generated by the algorithm and $\{\beta_k\}$ satisfies (2.52) for any convergent subsequence whose limit is not a critical point. Then either*

$$(2.59) \quad \lim_{k \rightarrow \infty} f(x_k) = -\infty,$$

or for every convergent subsequence $\{x_k\}_{k \in K}$ such that

$$(2.60) \quad \lim_{k \in K} \|x_{k+1} - x_k\| = 0,$$

we have

$$(2.61) \quad \lim_{k \in K} \|x_k - P[x_k - \nabla f(x_k)]\| = 0.$$

Proof. First of all we assume that there exists $M > -\infty$ such that

$$(2.62) \quad f(x_k) \geq M \quad \forall k;$$

otherwise, we would have the thesis.

Assume that (2.61) does not hold; therefore,

$$(2.63) \quad \lim_{k \in K} \|x_k - P[x_k - \nabla f(x_k)]\| \neq 0.$$

Then, as in the proof of Theorem 2.4, we can show that there exists $\alpha_1 > 0$ such that

$$(2.64) \quad \liminf_{k \in K} \|\nabla^1 f(x_k)\| = \alpha_1$$

and that the following holds:

$$(2.65) \quad \lim_{k \in K} \|d_k^1\| = 0.$$

Let us introduce the following notation:

$$I^1 = \left\{ i \in \overline{1, n} : \lim_{k \in K} \nabla_{x_i} f(x_k) \neq 0 \right\},$$

$$I_k^1 = I^1 \cap \{i \in \overline{1, n} : i \notin I_k^+\}.$$

Denote by

$$(2.66) \quad \min_{i \in I^1} \|\nabla_{x_i} f(\bar{x})\| = \alpha_2 > 0.$$

The first step in the proof is to show that for infinitely many $k \in K$

$$(2.67) \quad I_k^1 = I_{k+1}^1.$$

Assume that this is not true; thus for infinitely many $k \in K$ we have, for some $i \in I^1$, either

$$(2.68) \quad (x_k)_i > \varepsilon_k, \quad \text{or} \quad (x_k)_i \leq \varepsilon_k \quad \text{and} \quad \nabla_{x_i} f(x_k) \leq 0,$$

and

$$(2.69) \quad (x_{k+1})_i \leq \varepsilon_{k+1} \quad \text{and} \quad \nabla_{x_i} f(x_{k+1}) > 0,$$

or

$$(2.70) \quad (x_k)_i \leq \varepsilon_k \quad \text{and} \quad \nabla_{x_i} f(x_k) > 0,$$

and

$$(2.71) \quad (x_{k+1})_i > \varepsilon_{k+1}, \quad \text{or} \quad (x_{k+1})_i \leq \varepsilon_{k+1} \quad \text{and} \quad \nabla_{x_i} f(x_{k+1}) \leq 0.$$

Both cases (2.68)–(2.69), (2.70)–(2.71) can be analyzed in the same way; therefore we will only consider (2.68)–(2.69). We assume that there exists $i \in I^1$ such that

$$(2.72) \quad (x_k)_i > \varepsilon_k, \quad \nabla_{x_i} f(x_k) > 0,$$

$$(2.73) \quad (x_{k+1})_i \leq \varepsilon_{k+1}, \quad \nabla_{x_i} f(x_{k+1}) > 0$$

for sufficiently large $k \in K$ (because (2.60) holds and ∇f is continuous).

Due to our assumption (2.60), (2.63) and the definition of $\{\varepsilon_k\}$ we have

$$(2.74) \quad \lim_{k \in K} \varepsilon_k = \lim_{k \in K} \varepsilon_{k+1} = \lim_{k \in K} \min\{\varepsilon, \|x_k - P[x_k - D\nabla f(x_k)]\|\} = \bar{\varepsilon} > 0.$$

Furthermore, we have

$$(2.75) \quad \begin{aligned} (x_k)_i - (P[x_k - D\nabla f(x_k)])_i &= \begin{cases} (x_k)_i & \text{if } (x_k)_i - d_i \nabla_{x_i} f(x_k) \leq 0, \\ d_i \nabla_{x_i} f(x_k) & \text{if } (x_k)_i - d_i \nabla_{x_i} f(x_k) > 0 \end{cases} \\ &= \min\{(x_k)_i, d_i \nabla_{x_i} f(x_k)\} \end{aligned}$$

(where $d_i > 0$ is the i th diagonal element of D), and from (2.60), (2.72)–(2.75), and (2.66)

$$\begin{aligned} \lim_{k \in K} (x_k)_i &= \lim_{k \in K} (x_{k+1})_i = \lim_{k \in K} \varepsilon_k = \bar{\varepsilon}, \\ \lim_{k \in K} \min\{(x_k)_i, d_i \nabla_{x_i} f(x_k)\} &= \min\{(\bar{x})_i, d_i \nabla_{x_i} f(\bar{x})\} \\ &\geq \min\{\bar{\varepsilon}, d_i \alpha_2\} > 0 \end{aligned}$$

due to the fact that $i \in I^1$. But this is impossible because for infinitely many $k \in K$ $i \in I_{k+1}^+$ ((2.73)) and thus according to (2.60), (2.75) (with $D = I$), and Lemma 2.3

$$(2.76) \quad \lim_{k \in K} \min\{(x_k)_i, \nabla_{x_i} f(x_k)\} = 0.$$

In order to conclude the proof let us assume that condition (2.53) of Theorem 2.4, for the subsequence $\{x_{k+1}\}_{k \in K}$, does not hold. Thus, for every $\nu \in (0, 1)$ we have $\langle \nabla^1 f(x_{k+1}), d_k^1 \rangle > \nu \|\nabla^1 f(x_{k+1})\| \|d_k^1\|$ (a contradiction to condition (2.53)), provided that $0 < \lambda_{k+1} < 1$ and $k \in K$ are sufficiently large.

Since $\|x_{k+1} - x_k\| \rightarrow_{k \in K} 0$ and (2.64)–(2.67) hold, we have the relation

$$(2.77) \quad \begin{aligned} 0 < \alpha_1 \nu &\leq \liminf_{k \in K} \left\langle \nabla^1 f(x_{k+1}), \frac{d_k^1}{\|d_k^1\|} \right\rangle \\ &\leq \lim_{k \in K} (-\|d_k^1\|) = 0, \end{aligned}$$

where the last inequality follows from

$$\langle \nabla^1 f(x_k), d_k^1 \rangle \leq -\|d_k^1\|^2,$$

which holds because of (1.14) and the fact that $\nabla_{x_i} f(x_k)(d_k)_i = -(d_k)_i^2$ for $i \in I_k^+$.

Equation (2.77) is impossible; thus (2.63) cannot happen. $\square$

We have to mention that condition (2.60) plays an important role in the analysis of convergence of conjugate gradient algorithms which are based on the Polak–Ribière rule (e.g., [27]).

3. The globally convergent algorithm. In this section we examine the algorithm with the sequence $\{\beta_k\}$ defined by

$$(3.1) \quad \beta_k = \frac{\|\nabla^1 f((x_k^1, x_{k-1}^2))\|^2}{|\langle \nabla^1 f((x_k^1, x_{k-1}^2)) - \nabla^1 f(x_{k-1}), \nabla^1 f((x_k^1, x_{k-1}^2)) \rangle|},$$

where, as usual, by x^1 we mean a vector defined by indices not belonging to I_k^+ and the vector $\nabla^1 f((x_k^1, x_{k-1}^2))$ is as follows;

$$\nabla^1 f((x_k^1, x_{k-1}^2)) = \{\nabla_{x_i} f((x_k^1, x_{k-1}^2))\}_{i \notin I_k^+}.$$

In [25] we proved that $\{\beta_k\}$ constructed in this way guarantees that directions generated by the algorithm are equivalent to those generated by the Polak–Ribière algorithm, if directional minimization is exact and $I_k^+ = \emptyset \forall k$.

We can prove the following theorem.

THEOREM 3.1. *If the function f satisfies Assumption 1 then the algorithm gives*

$$\lim_{k \rightarrow \infty} f(x_k) = -\infty,$$

or for any convergent subsequence $\{x_k\}_{k \in K}$

$$(3.2) \quad \lim_{k \in K} \|x_k - P[x_k - \nabla f(x_k)]\| = 0$$

provided that

- (i) β_k is given by (3.1),
- (ii) there exists $M < \infty$ such that $\alpha_k \leq M \forall k$.

Proof. We have for formula (3.1)

$$\begin{aligned} \beta_k \|d_{k-1}^1\| &= \frac{\|\nabla^1 f((x_k^1, x_{k-1}^2))\|^2 \|d_{k-1}^1\|}{|\langle \nabla^1 f((x_k^1, x_{k-1}^2)) - \nabla^1 f(x_{k-1}), \nabla^1 f((x_k^1, x_{k-1}^2)) \rangle|} \\ &\geq \frac{\|\nabla^1 f((x_k^1, x_{k-1}^2))\|^2 \|d_{k-1}^1\|}{L \|\nabla^1 f((x_k^1, x_{k-1}^2))\| \|x_k^1 - x_{k-1}^1\|}. \end{aligned}$$

Because we have

$$|(P[x_{k-1} + \alpha_{k-1} d_{k-1}] - x_{k-1})_i| \leq \alpha_{k-1} |(d_{k-1})_i| \quad \forall i \in \overline{1, n},$$

thus

$$\begin{aligned} \|x_k^1 - x_{k-1}^1\| &= \|(P[x_{k-1} + \alpha_{k-1} d_{k-1}])^1 - x_{k-1}^1\| \\ &\leq \alpha_{k-1} \|d_{k-1}^1\| \leq M \|d_{k-1}^1\|, \end{aligned}$$

and eventually

$$(3.3) \quad \beta_k \|d_{k-1}^1\| \geq \frac{\|\nabla^1 f((x_k^1, x_{k-1}^2))\|}{LM}.$$

If $f(x_k) \geq \tilde{L} > -\infty$ then, from Lemma 2.3, we have

$$\lim_{k \in K} \|x_k - P[x_k + d_k]\| = 0$$

and this is equivalent to

$$(3.4) \quad \lim_{k \in K} \min\{(x_k)_i, -(d_k)_i\} = 0 \quad \forall i \in \overline{1, n}.$$

Moreover, we have

$$(P[x_k + \alpha_k d_k])_i - (x_k)_i = \begin{cases} -(x_k)_i & \text{if } (x_k + \alpha_k d_k)_i \leq 0, \\ \alpha_k (d_k)_i & \text{if } (x_k + \alpha_k d_k)_i > 0. \end{cases}$$

Therefore,

$$\lim_{k \in K} (x_k - P[x_k + \alpha_k d_k])_i = \lim_{k \in K} \min\{(x_k)_i, -\alpha_k (d_k)_i\}$$

and, because (3.4) and (ii) hold, we have

$$(3.5) \quad \lim_{k \in K} (x_k - P[x_k + \alpha_k d_k])_i = 0 \quad \forall i \in \overline{1, n}.$$

We know from Lemma 2.5 that only violation of condition (2.52) can destroy our convergence result (3.2). If we assume that $\{x_{k-1}\}_{k \in K}$ is bounded then using Lemma 2.3 and the condition (ii) we can show that

$$\lim_{k \in K} \|x_k - x_{k-1}\| = 0$$

in the same way we have shown (3.5). This implies

$$\liminf_{k \in K} \|\nabla^1 f((x_k^1, x_{k-1}^2))\| = \liminf_{k \in K} \|\nabla^1 f(x_k^1)\|,$$

and this result together with (3.3) ensures that condition (2.52) is fulfilled.

Therefore, in order to complete the proof we have to show that $\{x_{k-1}\}_{k \in K}$ is bounded. Indeed for any component i of x_{k-1} such that $\lim_{k \in K} (x_{k-1})_i = \infty$ we have $\lim_{k \in K} \alpha_{k-1} (d_{k-1})_i = -\infty$ because $\{(x_k)_i\}_{k \in K}$ is bounded. This implies that $\lim_{k \in K} (d_{k-1})(x_k - x_{k-1})_i = \infty$, due to the condition (ii). But this is not possible because according to (2.23) we would have $\lim_{k \in K} f(x_k) = -\infty$. Thus, $\{x_{k-1}\}_{k \in K}$ is bounded. $\square$

Remark 2. Formula (3.1) requires computation of two gradients $\nabla^1 f((x_k^1, x_{k-1}^2))$ and $\nabla f(x_k)$ at every iteration of the algorithm. For the formula

$$(3.6) \quad \beta_k = \frac{\|\nabla^1 f(x_k)\|^2}{|\langle \nabla^1 f(x_k) - \nabla^1 f(x_{k-1}), \nabla^1 f(x_k) \rangle|},$$

which can be evaluated by using only one gradient calculated at every iteration, we have also observed fast convergence of $\{x_k\}$ generated by the algorithm although, in this case, we have not been able to prove its convergence under assumptions stated in Theorem 3.1. In this case, in order to guarantee global convergence, the condition

$$(3.7) \quad \beta_k \|d_{k-1}^1\| \geq \nu_1 \|\nabla^1 f(x_k)\|,$$

where ν_1 is some small positive number, has to be satisfied at every iteration and d_k set to $-\nabla f(x_k)$ (β_k to 1), if (3.7) is violated.

The next result applies to problems which satisfy sufficient optimality conditions together with the strict complementarity condition. We show the important property

that the algorithm determines the set of bounds that are active at the solution in a finite number of iterations.

ASSUMPTION 2. *Let x^* be a point which satisfies necessary optimality conditions for problem (1.1), (2.2). Assume also that f is twice continuously differentiable and there exist $m_2 > m_1 > 0$ such that*

$$(3.8) \quad m_1 \|z\|^2 \leq z^T \nabla_{xx}^2 f(x^*) z \leq m_2 \|z\|^2$$

for all z satisfying

$$(3.9) \quad z \neq 0 : (z)_i = 0 \quad \forall i \in A(x^*),$$

where

$$(3.10) \quad A(x) := \{i \in \overline{1, n} : (x)_i = 0\}.$$

Moreover,

$$(3.11) \quad \nabla_{x_i} f(x^*) > 0 \quad \forall i \in A(x^*).$$

We can prove the following theorem.

THEOREM 3.2. *Let $\bar{x}$ be a point at which Assumption 2 is satisfied. Assume that $\{x_k\}$ is generated by the algorithm, the assumptions of Theorem 3.1 hold, and ε_k is calculated according to (2.13). Then there exists $\delta > 0$ such that if*

$$\|x_{\bar{k}} - \bar{x}\| \leq \delta$$

for some $\bar{k}$, then $\{x_k\}$ converges to $\bar{x}$ and

$$A(\bar{x}) = A(x_k) = I_k^+ \quad \forall k \geq \bar{k} + 1.$$

Proof. From the strict complementarity condition (3.11) we know that there exists $\delta_1 > 0$ such that

$$\nabla_{x_i} f(x_k) > 0, (P[x_k - D\nabla f(x_k)])_i = 0 \text{ for all } i \in A(\bar{x})$$

and all x_k satisfying $\|x_k - \bar{x}\| \leq \delta_1$. Thus, from the definition of ε_k and the fact that $\bar{x}$ is a critical point, there exists $0 < \delta_2 \leq \delta_1$ such that

$$(3.12) \quad (x_k)_i \leq \varepsilon_k \quad \forall i \in A(\bar{x})$$

and

$$(3.13) \quad (x_k)_i > \varepsilon_k \quad \forall i \notin A(\bar{x})$$

for all x_k satisfying $\|x_k - \bar{x}\| \leq \delta_2$.

Because $\nabla_{x_i} f(\bar{x}) > 0$ for $i \in A(\bar{x})$ and (3.12)–(3.13) hold, we have

$$(3.14) \quad I_k^+ = A(\bar{x}),$$

if $\|x_k - \bar{x}\| \leq \delta_2$.

Next, there exist $\bar{\varepsilon} > 0$, $0 < \delta_3 \leq \delta_2$ for which

$$(3.15) \quad (x_k)_i > \bar{\varepsilon} \quad \forall i \notin A(\bar{x})$$

if $\|x_k - \bar{x}\| \leq \delta_3$.

Moreover, from (2.14)–(2.15) and the definition of d_k^1 , the following relation holds:

$$\begin{aligned} \|d_k^1\|^2 &= \|d_k\|^2 - \|\nabla^2 f(x_k)\|^2 \\ &\leq \|\nabla f(x_k)\|^2 - \|\nabla^2 f(x_k)\|^2 \\ &= \|\nabla^1 f(x_k)\|^2 \end{aligned} \quad (3.16)$$

(where the inequality follows from (2.14)). Thus, from (3.14)–(3.15) and the fact that $\bar{x}$ is a critical point we know that for any $\hat{\varepsilon} > 0$ there exists $0 < \hat{\delta} \leq \delta_3$ such that

$$|(d_k)_i| \leq \hat{\varepsilon} \quad \forall i \notin A(\bar{x}) \quad (3.17)$$

if $\|x_k - \bar{x}\| \leq \hat{\delta}$. In particular, there exists $0 < \delta_4 \leq \delta_3$ such that

$$|(d_k)_i| \leq \bar{\varepsilon}/M \quad \forall i \notin A(\bar{x}) \quad (3.18)$$

if $\|x_k - \bar{x}\| \leq \delta_4$. Here, M is as specified in (ii) of Theorem 3.1.

Now we will show that there exists $0 < \delta_5 \leq \delta_4$ such that

$$\alpha_k^i \leq \alpha_k \quad \forall i \in I_k^+ \text{ if } \|x_k - \bar{x}\| \leq \delta_5. \quad (3.19)$$

Assume the contrary. From the directional minimization rule we have the inequality

$$\sum_{i: \alpha_k^i > \alpha_k} \nabla_{x_i} f(x_k + d_k(\alpha_k))(d_k)_i \geq -\eta \|d_k^-(\alpha_k)\|^2.$$

This implies that

$$L_1(x_k) + L_2(x_k) \geq -\eta(R_1(x_k) + R_2(x_k)), \quad (3.20)$$

where

$$\begin{aligned} L_1(x_k) &= - \sum_{i \in I_k^+: \alpha_k^i > \alpha_k} \nabla_{x_i} f(x_{k+1}) \nabla_{x_i} f(x_k), \\ L_2(x_k) &= \sum_{i \notin I_k^+: \alpha_k^i > \alpha_k} \nabla_{x_i} f(x_{k+1})(d_k)_i, \\ R_1(x_k) &= \sum_{i \in I_k^+: \alpha_k^i > \alpha_k} [\nabla_{x_i} f(x_k)]^2, \\ R_2(x_k) &= \sum_{i \notin I_k^+} [(d_k)_i]^2. \end{aligned}$$

Equations (3.12) and (3.14) and the strict complementarity condition imply that

$$(x_{k+1})_i \leq \max\{(x_k)_i - \alpha_k \nabla_{x_i} f(x_k), 0\} \leq (x_k)_i \leq \varepsilon_k \quad \forall i \in I_k^+ \quad (3.21)$$

if $\|x_k - \bar{x}\| \leq \delta_2$. Therefore, because $\alpha_k \leq M \quad \forall k$ and (3.14), (3.17), (3.21) hold, for any $\tilde{\varepsilon} > 0$ there exists $\tilde{\delta} > 0$ such that

$$\|x_{k+1} - x_k\| \leq \tilde{\varepsilon}, \quad (3.22)$$

$$R_1(x_k) \leq -L_1(x_k) + \tilde{\varepsilon}, \quad (3.23)$$

if $\|x_k - \bar{x}\| \leq \tilde{\delta}$.

Moreover, from (3.14), (3.17) we know that

$$(3.24) \quad |L_2(x_k)| \leq \tilde{\varepsilon} \quad \text{and} \quad R_2(x_k) \leq \tilde{\varepsilon}$$

if $\tilde{\delta}$ is sufficiently small.

Equations (3.20), (3.23), and (3.24) lead to a contradiction. It is sufficient to take

$$\tilde{\varepsilon} = 0.5(1 - \eta)/(2 + \eta) \min_{i \in A(\bar{x})} \|\nabla_{x_i} f(\bar{x})\|^2 > 0$$

(which is positive due to the strict complementarity condition). Then, we have

$$\tilde{\varepsilon} - R_1(x_k) + \tilde{\varepsilon} \geq -\eta R_1(x_k) - \eta \tilde{\varepsilon}.$$

Due to the contradiction to (3.19), for any $0 < \delta \leq \tilde{\delta}$, there exists x_k such that $\|x_k - \bar{x}\| \leq \delta$, $R_1(x_k)$ is positive and

$$\begin{aligned} 0 &< (1 - \eta)R_1(x_k) \leq (2 + \eta)\tilde{\varepsilon} \leq 0.5(1 - \eta) \min_{i \in A(\bar{x})} \|\nabla_{x_i} f(\bar{x})\|^2 \\ &\leq 0.7(1 - \eta) \min_{i \in A(\bar{x})} \|\nabla_{x_i} f(x_k)\|^2. \end{aligned}$$

Because of (3.14) this is not possible; thus (3.19) is true.

Since $\nabla_{x_i} f(\bar{x}) > 0 \ \forall i \in A(\bar{x})$, $\alpha_k \leq M \ \forall k$, from (3.14), (3.15), (3.18), and (3.19) there exists $0 < \delta_6 \leq \delta_5$ such that

$$(3.25) \quad A(\bar{x}) = A(x_{k+1}) \quad \text{if} \quad \|x_k - \bar{x}\| \leq \delta_6$$

and, due to (3.22), also

$$(3.26) \quad \|x_{k+1} - \bar{x}\| \leq \delta_4 \quad \text{if} \quad \|x_k - \bar{x}\| \leq \delta_6.$$

From (3.14) and (3.25)–(3.26) we have

$$A(\bar{x}) = I_{k+1}^+ = A(x_{k+1}) \quad \text{if} \quad \|x_k - \bar{x}\| \leq \delta_6.$$

Therefore, if $\|x_k - \bar{x}\| \leq \delta_6$ we have $\|x_{k+1} - \bar{x}\| \leq \delta_4$, $A(\bar{x}) = A(x_{k+1})$, and the $(k + 1)$ th iteration of the algorithm reduces to the iteration of the algorithm applied to the unconstrained optimization problem defined by active constraints at $\bar{x}$.

Now, because $\|d_k\| \leq \|\nabla f(x_k)\|$ and $\alpha_k \leq M$, we can recall Proposition 1.12 in [2]. According to this proposition there exists the neighborhood of $\bar{x}$ — $\mathcal{N} \subset \mathcal{B}(\bar{x}, \delta_6)$ — $= \{x : \|x - \bar{x}\| \leq \delta_6\}$ such that, if $x_{k+1} \in \mathcal{N}$ and $A(\bar{x}) = A(x_{k+1})$ then $x_{k+2} \in \mathcal{N}$ and, from (3.25), $A(\bar{x}) = A(x_{k+2})$. That means that if for some $\bar{k} \in K$

$$x_{\bar{k}} \in \mathcal{N}, \quad A(x_{\bar{k}}) = A(\bar{x}),$$

then

$$x_k \in \mathcal{N}, \quad A(x_k) = A(\bar{x}) \quad \forall k \geq \bar{k} \quad \text{and} \quad \lim_{k \rightarrow \infty} x_k = \bar{x}$$

(from Proposition 1.12 in [2]).

To complete the proof of the theorem we have to show that there exists $\bar{\delta} > 0$ such that if $\|x_k - \bar{x}\| \leq \bar{\delta}$, then $x_{k+1} \in \mathcal{N}$ and $A(x_{k+1}) = A(\bar{x})$. According to the arguments we used to prove (3.25)–(3.26) for any $\tilde{\delta} > 0$ there exists $\bar{\delta} > 0$ such that if

$\|x_k - \bar{x}\| \leq \bar{\delta}$, then $\|x_{k+1} - \bar{x}\| \leq \tilde{\delta}$ and $A(x_{k+1}) = A(\bar{x})$. If we choose $\tilde{\delta}$ small enough we have $\mathcal{B}(\bar{x}, \tilde{\delta}) \subset \mathcal{N}$ as required. $\square$

Let us now consider the quadratic case, i.e.,

$$(3.27) \quad f(x) = 1/2x^T Qx,$$

where Q is a positive definite matrix. We also assume that Assumption 2 holds. If the algorithm is applied to problem (1.1)–(1.2) with f defined by (3.27), then after a finite number of iterations active constraints at a solution are identified and our problem is reduced to a problem of unconstrained optimization.

First of all, in this case, we assume that an exact directional minimization is applied. It means that at every iteration we look for α_k , which is a solution to the problem

$$(3.28) \quad f(P[x_k + \alpha_k d_k]) = \min_{\alpha > 0} f(P[x_k + \alpha d_k]).$$

It is not difficult to see that if rule (3.28) is applied then conditions (2.23)–(2.24) are satisfied. If function f is defined by (3.27) then finding α_k is equivalent to calculating the generalized Cauchy point as described in [10]. After a finite number of iterations, l , active constraints at a solution are identified, and (3.28) reduces to a standard exact directional minimization of a smooth function.

Let us introduce the following notation:

$$\begin{aligned} d^0 &= d_l, \quad g^1 = \nabla f(x_{l+1}), \quad d^1 = d_{l+1}, \quad g^2 = \nabla f(x_{l+2}), \quad d^2 = d_{l+2}, \dots, \\ d^k &= d_{l+k}, \quad g^k = \nabla f(x_{l+k}), \dots \end{aligned}$$

We can show that the directions $\underline{d}^k = d^k / \gamma_k^2$ (where $\gamma_k^2 > 0$ are defined in [25], p. 452) satisfy the relations

$$\begin{aligned} \underline{d}^1 &= -g^1 + \frac{(g^1)^T Q \underline{d}^0}{(\underline{d}^0)^T Q \underline{d}^0} \underline{d}^0, \\ \underline{d}^k &= -g^k + \sum_{j=0}^{k-1} \frac{(g^k)^T Q \underline{d}^j}{(\underline{d}^j)^T Q \underline{d}^j} \underline{d}^j, \quad k = 2, \dots, n-1 \end{aligned}$$

(see section 1.3.4 in [2]).

Moreover, we can show that

$$\text{span}\{d^0, d^1, \dots, d^k\} = \text{span}\{d^0, g^1, \dots, g^k\}, \quad k = 1, \dots, n-1$$

(here, $\text{span}\{a_1, \dots, a_n\}$ is a linear space spanned by vectors $a_1, \dots, a_n$) and that

$$(g^k)^T d^j = 0, \quad j = 0, \dots, k-1$$

(see section 1.3.4 in [2]). From that we obtain

$$(g^k)^T g^j = 0, \quad j = 1, \dots, k-1.$$

Therefore, we can show that

$$(3.29) \quad \underline{d}^k = -g^k + \beta^k \underline{d}^{k-1} + \frac{(g^k)^T Q \underline{d}^0}{(\underline{d}^0)^T Q \underline{d}^0} \underline{d}^0,$$

where

$$\beta^k = \frac{\langle g^k, g^k - g^{k-1} \rangle}{\|g^{k-1}\|^2} = \frac{\|g^k\|^2}{\|g^{k-1}\|^2}, \quad k = 1, \dots, n-1.$$

If

$$(3.30) \quad (g^k)^T Q d^0 = 0, \quad k = 1, \dots, n-1,$$

then from the iteration the active constraints at a solution have been identified, and the algorithm behaves as a typical conjugate gradient algorithm applied to problem (1.1)–(1.2) with the variables $(x_k)_i$, $i \in A(\bar{x})$ fixed at zero. Therefore, under (3.30), our approach to problem (1.1)–(1.2) is fully justified.

If (3.30) is satisfied then we can show that

$$(3.31) \quad (g^1)^T g^0 = 0.$$

Conditions (3.30) and (3.31) are the measure of conjugacy of the directions generated by the algorithm. Because the algorithm is applied to nonquadratic functions and directional minimization is not exact we are interested in situations when these conditions are only approximately satisfied. This requirement is fulfilled because when

$$\langle \nabla^1 f(x_k) - \nabla^1 f(x_{k-1}), \nabla^1 f(x_k) \rangle \approx 0,$$

or, when

$$\langle \nabla^1 f(x_k), \nabla^1 f(x_{k-1}) \rangle \approx \|\nabla^1 f(x_k)\|^2,$$

then, as it is easily seen, the algorithm is restarted

$$d_k \approx -\nabla f(x_k).$$

4. Numerical experiments. In order to verify the effectiveness of our algorithm we have tested it on problems from the CUTE collection [4]. We only tried it on problems with dimension $n \geq 500$ because our algorithm should prove its effectiveness on problems with larger dimensions.

The algorithm has been implemented on a Sun SPARCstation 10 Model 51 (with 96Mb of main memory) in FORTRAN using *double precision* accuracy. We then compared our algorithm to the limited memory BFGS algorithm [8].

We used the directional minimization described in section 2 supplemented by a quadratic interpolation search. If the step from the previous iteration, for the normalized direction, did not satisfy conditions (2.23), (2.24) we performed a quadratic interpolation search based on the directional derivative and a value of the function calculated at this step.

Then we again checked conditions (2.23), (2.24); if they were not satisfied, we applied the algorithm described just after the proof of Lemma 2.2. We used the following values of parameters: $\mu = 0.0001$, $\eta = 0.9$. We applied the following parameters for ε_k defined in (2.13): $\varepsilon = 0.001$ and D a diagonal matrix with elements equal to 0.001. A value of ε is not crucial, but the entries of the matrix D should be small enough to avoid the situation when too many components of d_k are only defined by components of $\nabla f(x_k)$. We used $\{\beta_k\}$ defined by (3.6) and we set ν_1 in restarting

TABLE 4.1
Numerical results for problems from the CUTE collection, $n \leq 2000$.

Problem	LIT	IF	CPU	CPU (L-BFGS-B)	IF (L-BFGS-B)
	$\varepsilon_k = \min(0.001, \ w_k\)$			m=5	
BDEXP n = 1000	4	6	0.13	1.66	15
JNLBRNGA n = 1024	140	165	5.31	7.60	92
JNLBRNGB n = 1024	893	1093	33.50	34.70	424
LINVERSE n = 999	533	673	25.64	28.81	291
LINVERSE n = 1999	100	114	9.53	62.46	323
MCCORMCK n = 1000	20	39	1.09	0.81	15
NONSCOMP n = 1000	42	56	1.17	3.66	45
OBSTCLAE n = 1024	134	153	5.12	8.40	90
OBSTCLAL n = 1024	53	67	2.11	3.85	40
OBSTCLBL n = 1024	63	80	2.58	4.23	50
OBSTCLBM n = 1024	43	55	1.97	2.72	33
OBSTCLBU n = 1024	56	69	2.22	4.62	44
PROBPENL n = 500	1	3	0.03	0.05	3
TORSION1 n = 1024	74	90	2.89	3.70	43
TORSION2 n = 1024	72	87	2.77	5.51	61
TORSION3 n = 1024	37	47	1.59	1.45	23
TORSION4 n = 1024	44	63	1.85	3.77	49
TORSION6 n = 1024	26	42	1.27	2.04	30
Total CPU			100.77	180.04	

criterion (3.7) to 10^{-6} . The limited memory BFGS algorithm [8], which was used for the comparison, was run with $m = 5$ of the updates saved.

The results of our computations are shown in the following tables. LIT is the number of iterations, IF is the number of functions evaluations, which in most cases is equal to LIT for the L-BFGS-B code. Our stopping criterion was

$$|(P[x - \nabla f(x)])_i - (x)_i| \leq 10^{-5} \quad \forall i \in \overline{1, n}.$$

Results for JNLBRNGB problem from the CUTE collection are not presented for $n = 5625$, $n = 10000$, $n = 15625$ because both algorithms failed for these problems. CPU time in brackets, in Table 4.3, is the total CPU time spent on the evaluation of functions and their gradients.

TABLE 4.2
Numerical results for problems from the CUTE collection, $2000 < n \leq 10000$.

Problem	LIT	IF	CPU	CPU (L-BFGS-B)	IF (L-BFGS-B)
	$\varepsilon_k = \min(0.001 \ w_k\)$			m=5	
BDEXP n = 5000	3	5	0.64	6.30	15
JNLBRNGA n = 5625	266	318	67.67	118.70	188
JNLBRNGA n = 10 000	395	437	187.36	222.90	236
MCCORMCK n = 5000	30	56	9.02	6.95	15
MCCORMCK n = 10000	16	37	11.35	13.78	15
NONSCOMP n = 5000	45	61	6.58	22.82	51
NONSCOMP n = 10000	42	62	14.61	46.38	45
OBSTCLAE n = 5625	210	237	51.81	137.9	258
OBSTCLAE n = 10000	344	393	160.13	532.80	526
OBSTCLAL n = 5625	119	133	29.39	42.67	90
OBSTCLAL n = 10 000	158	180	73.42	107.40	124
OBSTCLBL n = 5625	153	175	38.05	62.46	120
OBSTCLBL n = 10000	189	215	87.15	216.10	212
OBSTCLBM n = 5625	124	144	31.35	41.88	83
OBSTCLBM n = 10000	116	137	55.12	109.90	111
OBSTCLBU n = 5625	162	189	40.92	46.18	90
OBSTCLBU n = 10000	192	228	91.48	130.80	135
TORSION1 n = 5476	211	251	53.51	62.70	125
TORSION1 n = 10000	240	283	114.45	166.70	171
TORSION2 n = 5476	139	165	34.91	88.07	168
TORSION2 n = 10000	141	156	63.89	328.50	281
TORSION3 n = 5476	153	170	37.08	20.72	51
TORSION3 n = 10000	152	170	70.29	66.07	84
TORSION4 n = 5476	143	170	37.41	86.94	182
TORSION4 n = 10000	232	273	110.38	272.80	296
TORSION6 n = 5476	67	86	17.86	54.50	123
TORSION6 n = 10000	98	131	51.78	187.80	223
Total CPU			1547.64	3200.72	

TABLE 4.3
Numerical results for problems from the CUTE collection, $n > 10000$.

Problem	LIT	IF	CPU	CPU (L-BFGS-B)	IF (L-BFGS-B)
	$\varepsilon_k = \min(0.001, \ w_k\)$			m=5	
JNLBRNGA n = 15625	340	403	266.28 (191.98)	469.10	332
OBSTCLAE n = 15625	290	340	225.60 (163.76)	1035.00	660
OBSTCLAL n = 15625	202	233	157.31 (113.26)	207.10	156
OBSTCLBL n = 15625	154	184	122.26 (88.73)	430.40	272
OBSTCLBM n = 15625	160	192	126.32 (92.28)	230.70	146
OBSTCLBU n = 15625	298	340	227.72 (164.39)	298.90	194
TORSION1 n = 14884	303	355	223.17 (164.15)	320.40	224
TORSION2 n = 14884	328	376	241.34 (175.65)	842.00	521
TORSION3 n = 14884	187	207	136.03 (97.79)	89.17	76
TORSION4 n = 14884	210	243	156.35 (113.15)	578.70	417
TORSION6 n = 14884	145	177	110.50 (81.03)	507.00	362
Total CPU			1992.88 (1446.17)	5008.47	

We also tested the algorithm with the ε_k defined by (2.26) with the parameters $\varepsilon_1 = 0.001$, $\varepsilon_2 = 1.00$ and the matrix D , a diagonal matrix with the entries equal to 0.001. The results for this version of our method were very similar to those stated in Tables 4.1–4.3. The corresponding cumulative CPU time for the problems in Table 4.1 was 96.62 while those for Tables 4.2–4.3 were 1686.65 and 2017.26, respectively. Cumulative numbers of iterations were 2339 (2335), 4436 (4140), and 2642 (2617), while cumulative numbers of functions evaluations were 2905 (2902), 5237 (4862), and 3127 (3050). (The numbers in brackets are the corresponding cumulative numbers for Tables 4.1–4.3.) Finally, in Table 4.4, we show the results for the version of the algorithm when $d_k^1 = -\nabla^1 f(x_k)$ at every iteration (we call this version the steepest descent: stp. descent). In this case the algorithm is a projection algorithm as stated in [1] except the directional minimization rule. The results for problems with $n < 2000$ (results for problems with $n > 2000$ were even more favorable for the conjugate gradient version of the algorithm) clearly show the effectiveness of our approach.

NOMENCLATURE.

- $\text{Nr}\{a, b\}$ – the point from a line segment spanned by the vectors a , b ,
which has the smallest norm,
 $P[x]$ – the projection operator of x on a specified convex set,
 α_k^i – the breakpoint for i th component of x_k
along a direction d_k ,

TABLE 4.4

Numerical results for problems from the CUTE collection, $n \leq 2000$. 900* means that the specified maximum number of iterations, 900, has been exceeded.

Problem	LIT	IF	CPU	CPU (stp. descent)	IF (stp. descent)
	$\varepsilon_k = \min(0.001, \ w_k\)$		$\varepsilon_k = \min(0.001, \ w_k\)$		
BDEXP n = 1000	4	6	0.13	0.12	6
JNLBRNGA n = 1024	140	165	5.31	40.58	900*
JNLBRNGB n = 1024	893	1093	33.50	32.29	900*
LINVERSE n = 999	533	673	25.64	5.4	128
LINVERSE n = 1999	100	114	9.53	130.48	900*
MCCORMCK n = 1000	20	39	1.09	1.45	45
NONSCOMP n = 1000	42	56	1.17	1.65	75
OBSTCLAE n = 1024	134	153	5.12	17.64	481
OBSTCLAL n = 1024	53	67	2.11	10.06	288
OBSTCLBL n = 1024	63	80	2.58	11.0	324
OBSTCLBM n = 1024	43	55	1.97	8.36	246
OBSTCLBU n = 1024	56	69	2.22	11.41	335
PROBPENL n = 500	1	3	0.03	0.05	3
TORSION1 n = 1024	74	90	2.89	22.47	635
TORSION2 n = 1024	72	87	2.77	32.25	935
TORSION3 n = 1024	37	47	1.59	5.93	166
TORSION4 n = 1024	44	63	1.85	10.92	305
TORSION6 n = 1024	26	42	1.27	3.64	100
Total CPU			100.77	345.7	

$$d_k(\alpha) - (d_k(\alpha))_i = \begin{cases} \alpha(d_k)_i & \text{if } \alpha \leq \alpha_k^i, \\ \alpha_k^i(d_k)_i & \text{if } \alpha > \alpha_k^i, \end{cases}$$

$\{t_k^i\}_1^m$ – an ordered sequence of breakpoints for a vector x_k

along a direction d_k ,

$$I_k^+ - \{i \in \overline{1, n} : (x_k)_i \leq \varepsilon_k \text{ and } \nabla_{x_i} f(x_k) > 0\},$$

$$d_{k-1}^+ - (d_{k-1}^+)_i = \begin{cases} (d_{k-1})_i & \text{if } i \notin I_k^+, \\ -\nabla_{x_i} f(x_k)/\beta_k & \text{if } i \in I_k^+, \end{cases}$$

$$I_k(\alpha) - \{i \in \overline{1, n} : i \notin I_k^+\} \bigcup \{i \in \overline{1, n} : i \in I_k^+ \text{ and } \alpha_k^i > \alpha\},$$

$$d_k^-(\alpha) - \{(d_k)_i\}_{i \in I_k(\alpha)},$$

$$\begin{aligned}
g_k(\alpha) - (g_k(\alpha))_i &= \begin{cases} \nabla_{x_i} f(x_k + \alpha d_k) & \text{if } \alpha < \alpha_k^i, \\ 0 & \text{if } \alpha \geq \alpha_k^i, \end{cases} \\
x^1 - \{x_i\}_{i \notin I_k^+}, \\
x^2 - \{x_i\}_{i \in I_k^+}, \\
\nabla^1 f(x) - \{\nabla_{x_i} f(x)\}_{i \notin I_k^+}, \\
d_{k-1}^1 - \{(d_{k-1})_i\}_{i \notin I_k^+}, \\
A(x) - \{i \in \overline{1, n} : x_i = 0\}.
\end{aligned}$$

Acknowledgments. The author thanks the associate editor and two anonymous referees for their helpful comments and suggestions.

REFERENCES

- [1] D. P. BERTSEKAS, *On the Goldstein–Levitin–Polyak gradient projection method*, IEEE Trans. Automat. Control, 21 (1974), pp. 174–184.
- [2] D. P. BERTSEKAS, *Constrained Optimization and Lagrange Multipliers Methods*, Academic Press, New York, 1982.
- [3] D. P. BERTSEKAS, *Projected Newton methods for optimization problems with simple constraints*, SIAM J. Control Optim., 20 (1982), pp. 221–245.
- [4] I. BONGARTZ, A. R. CONN, N. I. M. GOULD, AND Ph. L. TOINT, *CUTE: Constrained and Unconstrained Testing Environment*, Research report RC 18860, IBM T. J. Watson Research Center, Yorktown Heights, NY, 1994.
- [5] A. BUCKLEY AND A. LENIR, *QN-like variable storage conjugate gradients*, Math. Programming, 27 (1983), pp. 155–175.
- [6] J. V. BURKE AND J. J. MORE, *On the identification of active constraints*, SIAM J. Numer. Anal., 25 (1988), pp. 1197–1211.
- [7] J. V. BURKE AND J. J. MORE, *Exposing constraints*, SIAM J. Optim., 4 (1994), pp. 573–595.
- [8] R. H. BYRD, P. LU, J. NOCEDAL, AND C. ZHU, *A limited memory algorithm for bound constrained optimization*, SIAM J. Sci. Comput., 16 (1995), pp. 1190–1208.
- [9] F. H. CLARKE, *Generalized gradients and applications*, Trans. Amer. Math. Soc., 205 (1975), pp. 247–262.
- [10] A. R. CONN, N. I. M. GOULD, AND Ph. L. TOINT, *Global convergence of a class of trust region algorithms for optimization with simple bounds*, SIAM J. Numer. Anal., 28 (1988), pp. 433–460.
- [11] A. R. CONN, N. I. M. GOULD, AND Ph. L. TOINT, *Testing a class of methods for solving minimization problems with simple bounds on the variables*, Math. Comp., 50 (1988), pp. 399–430.
- [12] A. R. CONN, N. I. M. GOULD, AND Ph. L. TOINT, *LANCELOT: A Fortran package for large-scale nonlinear optimization (release A)*, in Springer Series in Computational Mathematics 17, Springer-Verlag, New York, 1992.
- [13] J. C. DUNN, *Gradient projection methods for systems optimization problems*, Control Dyn. Syst., 29 (1988), pp. 135–195.
- [14] R. FLETCHER, *Practical Methods of Optimization*, John Wiley, Chichester, 1987.
- [15] R. FLETCHER AND C. M. REEVES, *Function minimization by conjugate gradients*, Comput. J., 7 (1964), pp. 149–154.
- [16] J. CH. GILBERT AND C. LEMARÉCHAL, *Some numerical experiments with variable-storage Quasi-Newton algorithms*, Math. Programming, 45 (1989), pp. 407–436.
- [17] J. CH. GILBERT AND J. NOCEDAL, *Global convergence properties of conjugate gradient methods for optimization*, SIAM J. Optim., 2 (1992), pp. 21–42.
- [18] A. A. GOLDSTEIN, *Convex programming in Hilbert spaces*, Bull. Amer. Math. Soc., 70 (1964), pp. 709–710.
- [19] C. LEMARÉCHAL, *An extension of Davidon methods to nondifferentiable problem*, in Mathematical Programming Study 3, M. L. Balinski and P. Wolfe, eds., North-Holland, Amsterdam, 1975, pp. 95–109.
- [20] E. S. LEVITIN AND B. T. POLYAK, *Constrained minimization methods*, Zh. Vychisl. Mat. i Mat. Fiz., 6 (1965), pp. 787–823 (in Russian).

- [21] D. C. LIUA AND J. NOCEDAL, *On the limited memory BFGS method for large scale optimization problems*, Math. Programming, 45 (1989), pp. 503–528.
- [22] R. MIFFLIN, *An algorithm for constrained optimization with semismooth functions*, Math. Oper. Res., 2 (1977), pp. 191–207.
- [23] E. POLAK AND C. RIBIÉRE, *Note sur la convergence de methods de directions conjuges*, Rev. Francaise Informat. Recherche Operationnelle, 16 (1969), pp. 35–43.
- [24] R. PYTLAK, *Numerical experiment with new conjugate direction methods for nondifferentiable optimization*, in Proc. 28th IEEE CDC Conference, Tampa, FL, 1989, pp. 2457–2462.
- [25] R. PYTLAK, *On the convergence of conjugate gradient algorithms*, IMA J. Numer. Anal., 14 (1994), pp. 443–460.
- [26] D. F. SHANNO, *Conjugate gradient methods with inexact searches*, Math. Oper. Res., 3 (1978), pp. 244–256.
- [27] D. F. SHANNO, *On the convergence of a new conjugate gradient algorithm*, SIAM J. Numer. Anal., 15 (1978), pp. 1247–1257.
- [28] P. WOLFE, *A method of conjugate subgradients for minimizing nondifferentiable functions*, in Math. Programming Stud. 3, M. L. Balinski and P. Wolfe, eds., 1975, pp. 145–173.
- [29] C. ZHU, R. H. BYRD, P. LU, AND J. NOCEDAL, *L-BFGS-B—FORTRAN Subroutines for Large-Scale Bound Constrained Optimization*, Research report, Northwestern University, Department of Electrical Engineering and Computer Science, Evanston, IL, 1994.

COST APPROXIMATION: A UNIFIED FRAMEWORK OF DESCENT ALGORITHMS FOR NONLINEAR PROGRAMS*

MICHAEL PATRIKSSON[†]

Abstract. This paper describes and analyzes the cost approximation algorithm. This class of iterative descent algorithms for nonlinear programs and variational inequalities places a large number of algorithms within a common framework and provides a means for analyzing relationships among seemingly unrelated methods. A common property of the methods included in the framework is that their subproblems may be characterized by monotone mappings, which replace an additive part of the original cost mapping in an iterative manner; alternately, a step is taken in the direction obtained in order to reduce the value of a merit function for the original problem. The generality of the framework is illustrated through examples, and the convergence characteristics of the algorithm are analyzed for applications to nondifferentiable optimization. The convergence results are applied to some example methods, demonstrating the strength of the analysis compared to existing results.

Key words. nondifferentiable optimization, cost approximation, partial linearization, descent algorithms, convergence analysis

AMS subject classifications. 90C30, 49M37

PII. S105262349427577X

1. Introduction. Let $u : \mathbb{R}^n \mapsto \mathbb{R} \cup \{+\infty\}$ be lower semicontinuous (l.s.c.), proper, and convex and $f : \mathbb{R}^n \mapsto \mathbb{R} \cup \{+\infty\}$ be continuously differentiable on an open neighborhood of $\text{dom } u$. Consider the *nondifferentiable optimization problem*

$$[\text{NDP}] \quad \min_{x \in \mathbb{R}^n} T(x) := f(x) + u(x).$$

This problem is generic in mathematical programming and encompasses the problem of minimizing a convex and/or continuously differentiable real-valued function over a nonempty, closed, and convex set in $\mathbb{R}^n$.

The most common approach to solving [NDP] is to construct a sequence $\{x^t\}$ of iterates in $\text{dom } u$ such that the sequence $\{T(x^t)\}$ is strictly monotonically decreasing; typically, the sequence $\{x^t\}$ converges to a solution x^* to the *generalized equation* [Rob79]

$$[\text{GE}] \quad \nabla f(x^*) + \partial u(x^*) \ni 0,$$

which, under the mild assumption below (cf. [Roc66]), constitutes the *necessary* optimality conditions of x^* in [NDP] and further is *sufficient* for the global optimality of x^* in [NDP] whenever either f is convex or u is an indicator function of a nonempty, closed, and convex set and f is pseudoconvex.

The directional derivative of u in the direction of d is

$$u'(x; d) := \lim_{\lambda \downarrow 0} \frac{1}{\lambda} [u(x + \lambda d) - u(x)].$$

*Received by the editors October 17, 1994; accepted for publication (in revised form) February 22, 1997. This research was supported in part by grants from the Swedish Institute (303 GH/MLH) and the Swedish Research Council for Engineering Sciences (TFR) (282-93-1195).

<http://www.siam.org/journals/siopt/8-2/27577.html>

[†]Department of Mathematics, Chalmers University of Technology, S-412 96 Gothenburg, Sweden (mipat@math.chalmers.se). Formerly with Department of Mathematics, Linköping Institute of Technology, S-581 83 Linköping, Sweden.

Assumption 1.1 (lower semicontinuity of $u'(x; \cdot)$). For any $x \in \text{dom } u$ and $d \in \mathbb{R}^n$,

$$(1) \quad u'(x; d) = \liminf_{e \rightarrow d} u'(x; e).$$

Note that (1) is fulfilled for any $x \in \text{rint}(\text{dom } u)$ [Roc70a, Thm. 23.4]. A direct consequence of the assumption is that for any $x \in \text{dom } u$ and $d \in \mathbb{R}^n$ the directional derivative may be calculated as ([Roc66] and [Roc70a, Thm. 23.2])

$$(2) \quad u'(x; d) = \sup \{ \xi_u(x)^T d \mid \xi_u(x) \in \partial u(x) \}.$$

We let Ω denote the set of solutions to [GE], and assume that it is nonempty; a sufficient condition for this to be the case is that T is *coercive*, that is, $\text{dom } T$ is bounded or

$$(3) \quad \lim_{\|x\| \rightarrow \infty} \{T(x)/\|x\|\} = +\infty.$$

Moreover, Ω is a singleton set if it is nonempty and if T is strictly convex on its effective domain. Note that *strong convexity* of T , that is, there exists a constant $m_T > 0$ such that for any $\xi_T(x) \in \partial T(x)$ and $\xi_T(y) \in \partial T(y)$, $x, y \in \text{dom } T$,

$$(4) \quad [\xi_T(x) - \xi_T(y)]^T (x - y) \geq m_T \|x - y\|^2,$$

implies both strict convexity and coercivity of T and hence ensures the existence of a unique solution to [NDP]. (The expression (4) actually states that the subdifferential mapping ∂T is strongly monotone on $\text{dom } T$; we may use it as a definition of convexity of T by allowing $m_T = 0$.) Introducing some additional terms, a mapping Π is *maximal monotone* [Min62, Bro68] if it is monotone and its graph is not properly contained in the graph of any other monotone operator; its inverse mapping is denoted by Π^{-1} . Further, a mapping Π is *Lipschitz continuous* on $\text{dom } \Pi$ (with modulus $M_\Pi \geq 0$) if

$$\|\Pi(x) - \Pi(y)\| \leq M_\Pi \|x - y\| \quad \forall x, y \in \text{dom } \Pi.$$

An interesting application of [NDP] results from letting u be the *indicator function* of a nonempty, closed, and convex subset X of $\mathbb{R}^n$, that is, $u \equiv \delta_X$, where

$$\delta_X(x) := \begin{cases} 0, & x \in X, \\ +\infty, & x \notin X. \end{cases}$$

The subdifferential mapping of δ_X is the *normal cone* operator associated with the set X ,

$$(5) \quad N_X(x) := \begin{cases} \{z \in \mathbb{R}^n \mid z^T(y - x) \leq 0 \quad \forall y \in X\}, & x \in X, \\ \emptyset, & x \notin X. \end{cases}$$

(This is an example of a maximal monotone operator.) The problem [GE] then reduces to the problem of finding a vector x^* such that

$$\nabla f(x^*) + N_X(x^*) \ni 0,$$

which (without the need for Assumption 1.1) describes the first-order necessary conditions for the optimality of $x^* \in X$ in the constrained differentiable program

$$[\text{CDP}] \quad \min_{x \in X} f(x).$$

We also note that the problem [NDP] encompasses *dual* formulations of convex programs (e.g., [FHN96]).

A large amount of literature has been devoted to the study of computational methods for [NDP]. The present paper provides a framework of algorithms for [NDP] which includes many methods previously analyzed and provides a convergence analysis which unifies and improves upon existing ones. The analysis includes investigations of inexact solutions of the auxiliary problems, different step length rules, and convergence rate results.

2. The cost approximation algorithm. The term *cost approximation* (CA) was coined in the author's Ph.D. thesis [Pat93a] to describe a framework of descent algorithms for nonlinear programs and variational inequality problems. In this section we present the algorithm framework for the solution of [NDP]. The main idea is to iteratively approximate the cost mapping ∇f in order to obtain a problem which is more easily solved in the sense that a (possibly) nonmonotone mapping is replaced by a monotone one. (The term cost mapping is taken from applications of [GE] in equilibrium problems, where ∇f usually represents a cost vector.) The solution to this problem defines a search direction in which a step is taken to decrease the value of T .

2.1. The subproblem phase. Let $x \in \text{dom } u$. We introduce a continuous and monotone *cost approximating mapping* $\Phi : \text{dom } u \mapsto \mathbb{R}^n$. If the mapping ∇f is replaced by Φ , then the error made in the approximation is obviously $\Phi - \nabla f$. This error is taken into account by subtracting from Φ the fixed error term $\Phi(x) - \nabla f(x)$. The approximation may alternatively be interpreted as a fixation of the second term of the original cost $[\Phi + \partial u] + [\nabla f - \Phi]$ at x .

Thus, we arrive at a generalized equation in which a point y is sought such that

$$[\text{GE}_\Phi] \quad \Phi(y) + \partial u(y) + \nabla f(x) - \Phi(x) \ni 0.$$

We let $Y(x)$ denote the set of solutions y to $[\text{GE}_\Phi]$. (Note that a fixed term may be added to Φ without affecting this subproblem.)

By construction, the approximation is exact at x . This fact is important since it provides a termination criterion for the algorithm: if x solves the subproblem $[\text{GE}_\Phi]$, then it immediately follows that x also solves [GE]. The reverse is also true, and the subproblem thus provides a reformulation of [GE] as a fixed point problem in the (possibly point-to-set) mapping $x \mapsto Y(x)$.

Remark 2.1. The descent property of the search direction requires the mapping Φ to be monotone; this crucial property is, seemingly, not possible to relax within this framework.

In several known instances of CA algorithms, the corresponding mapping Φ may be identified as an approximation of ∇f (cf. Newton-type methods); the properties of ∇f in a given application may hence, implicitly through the requirements on Φ , restrict the possible choices of CA methods.

The convergence analysis provided in this paper concentrates on approximations made of the differentiable (and possibly nonconvex) function f and does not cover approximations of the convex function u , although such algorithms are easily devised within the framework; examples are subgradient optimization methods (e.g., [Sho85]) and the methods in [CoZ84]. Note that in such methods, however, the corresponding subproblems need not yield directions of descent with respect to T without proper modifications. $\square$

If Φ is chosen as the gradient mapping of a function $\varphi : \text{dom } u \mapsto \mathbb{R}$, which then is convex and continuously differentiable on $\text{dom } u$, then the subproblem $[\text{GE}_\Phi]$ reduces to solving the convex subproblem

$$[\text{NDP}_\varphi] \quad \min_{y \in \mathbb{R}^n} T_\varphi(y, x) := \varphi(y) + u(y) + f(x) - \varphi(x) + [\nabla f(x) - \nabla \varphi(x)]^T (y - x).$$

To give an example, assume that f is a convex function in C^2 on $\text{dom } u$ and choose $\varphi(y, x) := \frac{1}{2}(y - x)^T \nabla^2 f(x)(y - x)$, from which one obtains $T_\varphi(y, x) := u(y) + \{f(x) + \nabla f(x)^T (y - x) + \frac{1}{2}(y - x)^T \nabla^2 f(x)(y - x)\}$; an extension of Newton's method to nondifferentiable optimization is obtained, in which only the differentiable part of the objective is approximated. The choice $\varphi(y, x) := \frac{1}{2\gamma} \|y - x\|^2$, $\gamma > 0$, leads to extensions of steepest descent, gradient projection, and proximal point algorithms.

The construction of this subproblem may be given an alternative interpretation as a *partial linearization* [Pat93a] of T ; expressing it as $[\varphi + u] + [f - \varphi]$, the function T_φ is obtained from a first-order approximation (or linearization) of the second term at x .

Thus, by applying the CA concept, the problem $[\text{NDP}]$ can be solved through a sequence of convex optimization problems. Indeed, in most of the iterative methods for the solution of $[\text{NDP}]$ that one may identify as special cases from the class of CA algorithms, subproblems of the form $[\text{NDP}_\varphi]$ are solved. (Examples where the subproblem $[\text{GE}_\Phi]$ does not reduce to the convex optimization problem $[\text{NDP}_\varphi]$ include some matrix splitting methods for quadratic programs [CPS92] and Gauss–Seidel/SOR methods [OrR70].)

2.2. The line search phase. In general, we cannot expect the original problem to be solved by the subproblem solution y . An improved solution is therefore defined through a step taken in the direction of $d := y - x$ such that the value of a merit function for $[\text{NDP}]$ is reduced sufficiently. Although any merit function with sufficient continuity properties will do (such as the Euclidean distance to Ω), we choose T as the merit function. (In the solution of asymmetric variational inequality problems, *gap functions* associated with the CA subproblem $[\text{NDP}_\varphi]$ are available; see [Pat93c, LaP94, Pat94a, Pat94b, Pat97, Pat98] for further details.)

At the new point, the original mapping is again approximated—perhaps using another mapping Φ —and the algorithm proceeds until some stopping criterion is fulfilled.

2.3. The conceptual algorithm. A description of the CA algorithm is given in Table 1. A sequence $\{\Phi^t\}$ of monotone mappings is assumed to be given. (Note, however, that each mapping may also be constructed adaptively, given x^s , $s = 1, 2, \dots, t$, instead of being chosen a priori.)

It will be made clear in the rest of the paper how accurately the steps **1** and **3** need to be performed.

2.4. Conventions and assumptions. In the convergence analysis of the CA algorithm, three forms of the cost approximating mapping will be used. In the most general case, a sequence $\{\Phi^t\}$ of continuous and monotone mappings on $\text{dom } u$ is used; the mappings in this sequence are neither assumed to be dependent on each other nor to form a limit function (that is, equicontinuity is not assumed). In the second case, it is assumed that there exists a known continuous mapping $\Phi : \text{dom } u \times \text{dom } u \mapsto \mathbb{R}^n$ which is monotone on $\text{dom } u$ in its first argument; each individual cost approximating mapping Φ^t in the sequence $\{\Phi^t\}$ then has the form $\Phi(\cdot, x^t)$. (This form will be

TABLE 1
The CA algorithm.

-
0. (Initialization). Choose an initial point $x^0 \in \text{dom } u$, and let $t := 0$.
 1. (Search direction generation). Find a solution y^t to
- $$[\text{GE}_{\Phi^t}]$$
- $$(6) \quad \Phi^t(y^t) + \partial u(y^t) + \nabla f(x^t) - \Phi^t(x^t) \ni 0.$$
- The resulting search direction is $d^t := y^t - x^t$.
 2. (Termination criterion). If x^t solves $[\text{GE}_{\Phi^t}] \rightarrow \text{Stop } (x^t \in \Omega)$. Otherwise, continue.
 3. (Line search). Choose a step length, ℓ_t , such that $x^t + \ell_t d^t \in \text{dom } u$ and the value of T is reduced sufficiently.
 4. (Update). Let $x^{t+1} := x^t + \ell_t d^t$ and $t := t + 1$.
 5. (Termination criterion). If x^t is acceptable $\rightarrow \text{Stop}$. Otherwise, go to Step 1.
-

required whenever we do not wish to impose strong monotonicity conditions on Φ .) The third, and most simple, form of mapping to be used is the iteration independent form, that is, $\Phi^t \equiv \Phi$ for all t for a given continuous and monotone mapping $\Phi : \text{dom } u \mapsto \mathbb{R}^n$. (The latter form will only be used in the linear convergence Theorem 4.6 and its Corollary 5.1.) Whenever the iteration dependency of Φ^t is insignificant in the context, the superscript will be suppressed. (This applies especially to the technical lemmas in section 3.)

In general, the CA algorithm is only asymptotically convergent. (Finite convergence is obtained when the problem enjoys a *sharpness* property; see [Pat93c, Pat98] for further details.) Therefore, we shall implicitly presume that the sequence of iterates is infinite. Further, we do not study the behavior of the algorithm when a solution does not exist.

The remainder of the paper is organized as follows. In the next section, we provide some properties of the search directions and the step length rules considered for different assumptions on the mappings Φ^t . In section 4, a convergence analysis is made. Some consequences of this analysis are described in section 5, demonstrating its strength compared to previous analyses of algorithms included in the framework.

3. Preliminaries.

3.1. Analysis of the search directions.

LEMMA 3.1 (characterizations of the search directions). *Let $x \in \text{dom } u$, $Y(x)$ be the (possibly empty) set of solutions to $[\text{GE}_{\Phi}]$, $y \in Y(x)$, and $d := y - x$.*

(a) (A fixed point characterization of Ω).

(1) $x \in \Omega \iff x \in Y(x)$.

(2) Let $\Phi \equiv \nabla \varphi$. Then, $x \in \Omega \iff x \in Y(x) \iff T_{\varphi}(y, x) = T_{\varphi}(x, x)$.

(b) (A characterization of $Y(x)$). Let Φ be maximal monotone. Then, $\Phi + \partial u$ is maximal monotone and

$$(7) \quad Y(x) = [\Phi + \partial u]^{-1}[\Phi - \nabla f](x).$$

(c) (A well-posedness characterization).

(1) Let Φ be maximal monotone. Then, the set $Y(x)$ is nonempty and locally bounded if

$$(8) \quad \Phi(x) - \nabla f(x) \in \text{int}(\text{dom}([\Phi + \partial u]^{-1})).$$

(2) Let $\Phi \equiv \nabla\varphi$. Then, the set $Y(x)$ is nonempty and bounded if and only if

$$(9) \quad \nabla\varphi(x) - \nabla f(x) \in \text{int}(\text{dom}([\varphi + u]^\circ)),$$

where $[\varphi + u]^\circ$ is the conjugate function of $\varphi + u$.

(d) Let $\Phi : \text{dom } u \times \text{dom } u \mapsto \mathbb{R}^n$ be a continuous mapping on $\text{dom } u \times \text{dom } u$ of the form $\Phi(y, x)$ and monotone on $\text{dom } u$ with respect to y . Let $S \subseteq \text{dom } u$ be an arbitrary, closed set in $\mathbb{R}^n$. Then, the mapping $x \mapsto D(x) := Y(x) - x$ is closed on S .

(e) Let u be strictly convex on $\text{dom } u$ or Φ strictly monotone on $\text{dom } u$. If $Y(x)$ is nonempty, then it is a singleton.

Proof.

(a) (1) Follows immediately from the construction of $[\text{GE}_\Phi]$.

(2) The second equivalence follows immediately from the construction of $[\text{NDP}_\varphi]$.

(b) The first result follows from [Roc70b, Roc70c]; the second result then follows by direct calculation.

(c) (1) The set $Y(x)$ is characterized by (7). Let $\Pi := \Phi + \partial u$. Since, by definition, $\text{range } \Pi = \text{dom } \Pi^{-1}$, the set $Y(x)$ is nonempty if and only if $[\Phi - \nabla f](x) \in \text{dom } \Pi^{-1}$; further, it is locally bounded if $[\Phi - \nabla f](x) \in \text{int}(\text{dom } \Pi^{-1})$ [Roc69].

(2) From [Roc70a, Cor. 23.5.1 and Thm. 27.1.b], the set $Y(x)$ is characterized by $Y(x) = \partial([\varphi + u]^\circ)(\nabla\varphi(x) - \nabla f(x))$. By [Roc70a, Thms. 23.4 and 27.1.d], $Y(x)$ is nonempty and bounded if and only if (9) holds.

(d) Let $\{x^t\} \subseteq S$ and $\{y^t\}$ be sequences in $\text{dom } u$ with $y^t \in Y(x^t)$, $\{x^t\} \rightarrow x$ and $\{y^t\} \rightarrow y$. By the continuity of Φ and ∇f and the convexity of u , Theorem 24.4 of [Roc70a] yields that $y \in Y(x)$. Hence, the mapping $x \mapsto Y(x)$ is closed on S . The graph of the mapping D is obtained from that of Y through an affine transformation; it then follows from [Ber63, p. 111] that $x \mapsto D(x)$ is closed on S , as desired.

(e) Follows from the strict monotonicity of $\Phi + \partial u$. $\square$

Remark 3.1. The fixed point property in (a) validates the termination criterion of Step 2 of the CA algorithm; the result (a)(2) shows that it is also easily checked whenever Φ is a gradient mapping.

The characterization (7) of the set $Y(x)$ made in (b) bears resemblance to iterative formulas describing proximal point methods [Mar70, Roc76] and forward-backward splitting methods [Bre73, EcB92] for finding a zero of the sum of two maximal monotone mappings. The class of CA algorithms is actually far more general; for example, the mapping Φ is not necessarily affine or strongly monotone and can be chosen to adapt to problem structures (such as separability in u and f), and a (possibly inexact) line search is embedded in the algorithm. We also note that a monotone and single-valued mapping on $\mathbb{R}^n$ is automatically maximal monotone.

The properties of $Y(x)$ characterized in the result (c) are natural well-posedness requirements on the subproblem $[\text{GE}_\Phi]$. A sufficient condition for (8) to hold is that $\Phi + \partial u$ is *weakly coercive* [Zei90, Def. 32.34.b], since this condition implies that $\text{dom}([\Phi + \partial u]^{-1}) = \mathbb{R}^n$ (see [Zei90, Cor. 32.35] and [Bro68]). The corresponding sufficient condition in the case where Φ is a gradient is that $\varphi + u$ is *coercive* (cf. (3)), since $\varphi + u$ then is *cofinite* [Roc70a, pp. 116 and 259] and $\text{dom}([\varphi + u]^\circ) = \mathbb{R}^n$ holds [Roc70a, Cor. 13.3.1]. All these conditions are satisfied if $\text{dom } u$ is bounded; an

important special case is when u includes, as an additive term, an indicator function of a compact set.

A function $\varphi : \text{dom } u \times \text{dom } u \mapsto \mathbb{R}$ of the form $\varphi(y, x)$, continuous on $\text{dom } u \times \text{dom } u$ and convex and differentiable on $\text{dom } u$ with respect to y , automatically has the continuity property of the mapping $\Phi = \nabla_y \varphi$ required in the result (d). Note that $\text{dom } u$ is not closed in general [Roc70a, p. 52], in which case it cannot take the role of the set S . $\square$

LEMMA 3.2 (descent properties). *Let $x \in \text{dom } u \setminus \Omega$, $Y(x)$ be the (possibly empty) set of solutions to $[\text{GE}_\Phi]$, $y \in Y(x)$, and $d := y - x$.*

(a) *If either u is strictly convex on $\text{dom } u$ and $\partial u(x)$ is bounded, or Φ is strictly monotone on $\text{dom } u$, then $T'(x; d) < 0$.*

(b) *Let $\Phi \equiv \nabla \varphi$, $\bar{y} \in \mathbb{R}^n$ be any point such that $T_\varphi(\bar{y}, x) < T_\varphi(x, x)$, and $\bar{d} := \bar{y} - x$. Then, $T'(x; \bar{d}) < 0$. Especially, $T'(x; d) < 0$.*

(c) *Let u be strongly convex on $\text{dom } u$ or Φ strongly monotone on $\text{dom } u$. Let $\bar{y} \in \mathbb{R}^n$ be an approximate solution to $[\text{GE}_\Phi]$ in the sense that for some vector $r \in \mathbb{R}^n$,*

$$(10) \quad \Phi(\bar{y}) + \partial u(\bar{y}) + \nabla f(x) - \Phi(x) \ni r,$$

and $\bar{d} := \bar{y} - x$. If $\|r\| < (m_u + m_\Phi)\|\bar{d}\|$, then $T'(x; \bar{d}) < 0$. Especially, $Y(x)$ is nonempty and singleton and

$$(11) \quad T'(x; d) \leq -(m_u + m_\Phi)\|d\|^2.$$

(d) *Let Φ be strongly monotone on $\text{dom } u$ and Lipschitz continuous on $\text{dom } u$. Then, there exists a $\xi_u(y) \in \partial u(y)$ such that*

$$(12) \quad -\frac{T'(x; d)}{\|\nabla f(x) + \xi_u(y)\| \cdot \|d\|} \geq \rho,$$

where $\rho := m_\Phi/M_\Phi$.

Remark 3.2. The reader should note here that in (11) we make use of the convention that if a mapping Π is monotone, then it satisfies (4) with $m_\Pi = 0$. $\square$

Proof.

(a) If u is strictly convex, then for all $\xi_u(x) \in \partial u(x)$, $[-\Phi(y) - \nabla f(x) + \Phi(x) - \xi_u(x)]^T d > 0$ holds, which, from the monotonicity of Φ , yields

$$(13) \quad [\nabla f + \xi_u(x)]^T d < 0 \quad \forall \xi_u(x) \in \partial u(x).$$

If $\partial u(x)$ is bounded, then the supremum of the left-hand side of the above inequality over $\xi_u(x) \in \partial u(x)$ is attained, and the result follows.

If Φ is strictly monotone, then the monotonicity of ∂u [Roc70a, Cor. 31.5.2] yields $[\nabla f + \xi_u(x)]^T d \leq [\Phi(x) - \Phi(y)]^T d < 0$, which implies the desired result.

(b) By the convexity of φ , $T_\varphi(\bar{y}, x) < T_\varphi(x, x)$ implies that $\nabla f(x)^T \bar{d} < u(x) - u(\bar{y})$. Then, from (2),

$$(14) \quad \begin{aligned} T'(x; \bar{d}) &= \nabla f(x)^T \bar{d} + \sup\{\xi_u(x)^T(\bar{y} - x) \mid \xi_u(x) \in \partial u(x)\} \\ &< u(x) - u(\bar{y}) + \sup\{\xi_u(x)^T \bar{d} \mid \xi_u(x) \in \partial u(x)\}, \end{aligned}$$

and the right-hand side of (14) is nonpositive since u is convex. The point y is just a special case of such a $\bar{y}$.

(c) We obtain from (10) that

$$[\nabla f(x) + \xi_u(x)]^T \bar{d} \leq -(m_u + m_\Phi) \|\bar{d}\|^2 + \|r\| \cdot \|\bar{d}\|$$

holds for all $\xi_u(x) \in \partial u(x)$, from which the result easily follows.

(d) By $[\text{GE}_\Phi]$ and the Lipschitz continuity of Φ , there exists a $\xi_u(y) \in \partial u(y)$ such that $\|\nabla f(x) + \xi_u(y)\| \leq M_\Phi \|d\|$. Together with (11), with $m_u = 0$, (12) follows. $\square$

Remark 3.3. The second result of (a) requires that Φ is strictly monotone if it is not a gradient mapping (cf. the result (b)); that nonstrict monotonicity is not sufficient is clear from the following example: take $u := \delta_X$, where $X := \{x \in \mathbb{R}^2 \mid 0 \leq x \leq 2\}$, and $f(x) := \frac{1}{2}(x_1^2 + x_2^2)$. Let further $x := (2, 2)^T$, which clearly is nonoptimal, and

$$\Phi(x) := \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} x,$$

which is a nonstrictly monotone and asymmetric mapping. By simple calculations, $y = (0, 4)^T$ can be shown to solve $[\text{GE}_\Phi]$, but $\nabla f(x)^T(y - x) = 0$.

The result (b) establishes that descent properties hold under very mild assumptions if Φ is a gradient mapping and validates inexact solutions of $[\text{NDP}_\varphi]$. (In the above example, choose, for example, $\Phi := 0$; then $y = (0, 0)^T$ solves $[\text{GE}_\Phi]$, and $\nabla f(x)^T(y - x) = -8$.)

The result (c) establishes the descent property under conditions on the accuracy of the solution of $[\text{GE}_\Phi]$. While solving $[\text{GE}_\Phi]$ with some iterative procedure, the validity of (10) for a given value of r can often be checked easily.

The vector r appearing in (10) need not be associated only with a measure of the distance from $\bar{y}$ to $Y(x)$. The vector can include (or be equal to) an error in the calculation of $\nabla f(x)$; the result (c) thus provides conditions for obtaining descent when the value of $\nabla f(x)$ is observed in the presence of noise. See [Pol87, Chap. 4] for discussions on such errors and their effect on gradient methods.

The inequality (12) in (d) generalizes the property of *gradient relatedness* of search directions used in unconstrained differentiable optimization [OrR70, Def. 14.3.1]. $\square$

3.2. Step length rules. This subsection presents and validates the step length rules considered in the realization of Step 3 of the CA algorithm. Four different step length rules are used; the first is a conceptual one.

DEFINITION 3.3 (exact line search). *Let $x \in \text{dom } u$, $y \in Y(x)$, and $d := y - x$.*

(E) *Choose ℓ as a solution to $\min \{T(x + \ell d) \mid \ell \geq 0\}$.*

We next define a new step length rule, which generalizes the Armijo [Arm66] rule from differentiable optimization. The original statement of the Armijo rule cannot be applied here due to the nondifferentiability of T ; the fact that the generalized gradient cannot be Lipschitz continuous makes it necessary to replace the directional derivative used in the original Armijo rule with a monotonicity measure in Φ . Also, the unit step is the largest for which the modified Armijo rule can be validated (see Lemma 3.5 below); we therefore choose a unit first trial step length. Further, Φ must be chosen strictly monotone.

DEFINITION 3.4 (Armijo step length). *Let Φ be strictly monotone on $\text{dom } u$. Let $x \in \text{dom } u \setminus \Omega$, $Y(x) := \{y\}$, and $d := y - x$. Further, let $\alpha, \beta \in (0, 1)$.*

(A') *Let $\ell := \beta^{\bar{i}}$, where $\bar{i}$ is the smallest nonnegative integer i such that*

$$(15) \quad T(x + \beta^{\bar{i}} d) - T(x) \leq \alpha \beta^{\bar{i}} [\Phi(x) - \Phi(y)]^T d.$$

To see that this step length rule generalizes the Armijo rule, let $u \equiv 0$ in [NDP]. Then a subproblem solution y is characterized by the system of nonlinear equations $\nabla f(x) = \Phi(x) - \Phi(y)$ (cf. (6)); using this relationship in (15) yields that Rule A' reduces to the original Armijo rule, with a unit first trial step. (Note that Rule A' uses an over-estimating model of T , since $T'(x; d) \leq [\Phi(x) - \Phi(y)]^T d$ holds.)

The Armijo Rule A' is validated below.

LEMMA 3.5 (validity of the Armijo rule). *Let Φ be strictly monotone on $\text{dom } u$. Let $x \in \text{dom } u \setminus \Omega$, $Y(x) := \{y\}$, and $d := y - x$. Further, let $\alpha, \beta \in (0, 1)$.*

(a) *There exists a nonnegative integer $\bar{i}$ satisfying (15).*

(b) *Assume further that ∇f is Lipschitz continuous on $\text{dom } u$. Then, either a unit step satisfies (15) or*

$$(16) \quad \beta^{\bar{i}} > 2\beta(\alpha - 1) \frac{[\Phi(x) - \Phi(y)]^T d}{M_{\nabla f} \|d\|^2}.$$

Proof.

(a) Taylor's formula yields, for $\ell \in [0, 1]$,

$$(17) \quad f(x + \ell d) - f(x) = \ell \nabla f(x)^T d + o(\ell).$$

Further, for $\ell \in [0, 1]$ (cf. [Roc70a, Cor. 24.2.1 and Thm. 24.1], [Roc66]),

$$(18) \quad \begin{aligned} u(x + \ell d) - u(x) &= \int_0^\ell u'(x + sd; d) ds \leq -\ell u'(x + d; -d) \\ &= \ell \inf \{ \xi_u(y)^T d \mid \xi_u(y) \in \partial u(y) \} \\ &\leq \ell \xi_u(y)^T d \quad \forall \xi_u(y) \in \partial u(y). \end{aligned}$$

In (18), let $\xi_u(y) \in \partial u(y)$ be given by (6). Then, combining (17) and (18),

$$T(x + \ell d) - T(x) \leq \ell [\Phi(x) - \Phi(y)]^T d + o(\ell).$$

For any value of $\alpha \in (0, 1)$, there must therefore be an $\bar{\ell} > 0$ such that, for all $\ell \in (0, \bar{\ell})$, $T(x + \ell d) - T(x) \leq \alpha \ell [\Phi(x) - \Phi(y)]^T d$ holds, from which we get the result.

(b) We begin by showing that for $\ell \in [0, 1]$,

$$(19) \quad T(x + \ell d) - T(x) \leq \ell [\Phi(x) - \Phi(y)]^T d + \frac{M_{\nabla f}}{2} \ell^2 \|d\|^2$$

holds. Taylor's formula and the Lipschitz continuity of ∇f yields, for $\ell \in [0, 1]$,

$$(20) \quad \begin{aligned} f(x + \ell d) - f(x) &= \int_0^\ell \nabla f(x + sd)^T d ds \leq \ell \nabla f(x)^T d + \int_0^\ell M_{\nabla f} \|d\|^2 s ds \\ &= \ell \nabla f(x)^T d + \frac{M_{\nabla f}}{2} \ell^2 \|d\|^2. \end{aligned}$$

Combining (20) with (18), the latter obtained as in (a) from $[\text{GE}_{\Phi^*}]$, we obtain (19). Using (19) and the strict monotonicity of Φ , by choosing $\ell \leq 2(\alpha - 1)([\Phi(x) - \Phi(y)]^T d) / (M_{\nabla f} \|d\|^2)$, we obtain

$$(21) \quad T(x + \ell d) - T(x) \leq \alpha \ell [\Phi(x) - \Phi(y)]^T d.$$

Replacing ℓ with $\beta^{\bar{i}}$, we see that $\bar{i}$ is the smallest integer i to satisfy

$$(22) \quad \beta^i \leq 2(\alpha - 1) \frac{[\Phi(x) - \Phi(y)]^T d}{M_{\nabla f} \|d\|^2}.$$

From (22) then follows the desired result. $\square$

Remark 3.4. From (22) we may find conditions under which the unit step, that is, $\bar{i} = 0$, is accepted by Rule A'. Assume that Φ is strongly monotone on $\text{dom } u$. Then, a unit step is obtained if

$$(23) \quad \frac{2(1 - \alpha)m_\Phi}{M_{\nabla f}} \geq 1.$$

Note that, from (16), the step length given by Rule A' always satisfies $\ell \geq \min\{1, 2\beta(1 - \alpha)([\Phi(y) - \Phi(x)]^T d)/(M_{\nabla f} \|d\|^2)\}$, which reduces to

$$(24) \quad \ell \geq \min \left\{ 1, \frac{2\beta(1 - \alpha)m_\Phi}{M_{\nabla f}} \right\}$$

whenever Φ is strongly monotone. $\square$

If the calculation of T is expensive, then a step length formula which does not require its calculation is of interest. Such a rule is described next.

DEFINITION 3.6 (relaxation step length). *Assume that ∇f is Lipschitz continuous on $\text{dom } u$, and let Φ be strongly monotone on $\text{dom } u$. Further, let $0 < \varepsilon_i < m_\Phi/M_{\nabla f}$, $i = 1, 2$.*

(R) *Choose ℓ in the closed interval $(\varepsilon_1, \min\{1, 2m_\Phi/M_{\nabla f} - \varepsilon_2\})$.*

Remark 3.5. The inverse of the quotient $m_\Phi/M_{\nabla f}$ appearing in the description of the maximal step length is in some cases an approximate condition number of the Hessian of f ; consider, for example, Newton's method. $\square$

The step length rule is validated below.

LEMMA 3.7 (validity of the relaxation step). *Assume that ∇f is Lipschitz continuous on $\text{dom } u$, and let Φ be strongly monotone on $\text{dom } u$. Let $x \in \text{dom } u \setminus \Omega$, $Y(x) := \{y\}$, and $d := y - x$. If ℓ is chosen according to Rule R, then $T(x + \ell d) < T(x)$.*

Proof. The inequality (19), together with the strong monotonicity of Φ , yields, for all $\ell \in [0, 1]$, that

$$(25) \quad T(x + \ell d) - T(x) \leq \ell \left(-m_\Phi + \frac{M_{\nabla f}}{2} \ell \right) \|d\|^2,$$

from which the result easily follows. $\square$

Note that if

$$(26) \quad \frac{2m_\Phi}{M_{\nabla f}} > 1$$

holds, then a unit step yields descent. (This condition is implied by (23).)

The relaxation step length has the advantage of being very simple, but it requires the knowledge of (an estimate of) the Lipschitz constant of ∇f . (In certain cases, such as when f is quadratic or separable, estimates of the Lipschitz constant $M_{\nabla f}$ are available; see, for example, [WoZ96]. The maximal step length is usually determined through experiments.) The last step length rule introduced yields descent *eventually* and does not require the knowledge (or estimation) of any problem or method parameters.

DEFINITION 3.8 (divergent series step length).

(D) Let the sequence $\{\ell_t\}$ satisfy

$$(27) \quad [0, 1] \supset \{\ell_t\} \rightarrow 0, \quad \sum_{t=0}^{\infty} \ell_t = \infty.$$

Remark 3.6. The requirements of the relaxation and divergent series step length rules need not be viewed as defining step length formulas per se, but can instead be viewed as conditions that some arbitrary step length rule must satisfy in order to yield convergence. The *almost complete relaxation* strategy described in [DeV85, Sec. 3.4], for example, allows for the utilization of an arbitrary step length selection rule through the (almost never used) upwards or downwards rounding of a tentative step length in order to fall between two step lengths (very small and very large, respectively) defined by preselected formulas both satisfying (27). $\square$

4. Convergence results.

4.1. The conceptual method. In our first convergence result, we will assume that the mappings Φ^t are *monotone* only. In order to ensure the convergence of the CA algorithm, this will require that the sequence $\{\Phi^t\}$ is constructed by a given function φ on $\text{dom } u \times \text{dom } u$ (that is, that $\Phi^t \equiv \nabla\varphi(\cdot, x^t)$ for each t), and that the line search is performed exactly.

THEOREM 4.1 (convergence under Rule E). *Assume that u is continuous on $\text{dom } u$. Let the sequence $\{\Phi^t\}$ of cost approximating mappings be constructed such that, for each t , $\Phi^t \equiv \nabla\varphi(\cdot, x^t)$ for a given function $\varphi : \text{dom } u \times \text{dom } u \mapsto \mathbb{R}$ of the form $\varphi(y, x)$, continuous on $\text{dom } u \times \text{dom } u$, and convex and in C^1 on $\text{dom } u$ with respect to y . Assume that $x^0 \in \text{dom } u$ is such that the lower level set $L(x^0) := \{x \in \text{dom } u \mid T(x) \leq T(x^0)\}$ is bounded, and further that the problem $[\text{NDP}_\varphi]$ is well defined in the sense that (9) holds for every $x \in L(x^0)$. Let Rule E be used. Then, $\{T(x^t)\} \rightarrow T(\bar{x})$ for some $\bar{x} \in \Omega$, any accumulation point of the sequence $\{x^t\}$ (at least one such point exists) lies in Ω , and*

$$(28) \quad \left\{ \inf_{x \in \Omega \cap L(x^0)} \|x^t - x\| \right\} \rightarrow 0.$$

Proof. In order to apply Zangwill's Theorem A [Zan69, Sec. 4.5], we first identify the *solution set* with Ω , the *descent function* with T , and the *algorithmic map* with the composite mapping $A := ED$, where D is the direction finding mapping and E is the exact line search map. To fulfill the assumptions of Theorem A, we next show that (1) the sequence $\{x^t\}$ lies in a compact set, (2) $T(x^{t+1}) < T(x^t)$ if $x^t \notin \Omega$ and the algorithm terminates if $x^t \in \Omega$, and (3) the mapping A is closed at all points in $L(x^0) \setminus \Omega$.

- (1) From the boundedness assumption and [StW70, p. 134], the set $L(x^0)$ is compact; the descent property then ensures that the sequence $\{x^t\}$ lies in a compact set.
- (2) The descent property follows from Lemma 3.2(b). The termination property ($x^t \in \Omega$) follows from Lemma 3.1(a)(2).
- (3) The mapping D is closed on $L(x^0)$ by Lemma 3.1(d) (identifying S with $L(x^0)$), and the same is true for the mapping E , since T is continuous on $\text{dom } u$ [Zan69, Lem. 5.1]. The mapping $\partial([\varphi(\cdot, x) + u]^\circ)$ is monotone, and therefore, by (9) and [Zei90, Prop. 32.33], locally bounded at $\nabla\varphi(x, x) -$

$\nabla f(x)$. From (7), the mapping Y is therefore locally bounded on $L(x^0)$. From (1), the sequence $\{x^t\}$ is bounded. The local boundedness of Y implies that the sequence $\{y^t\}$ is also bounded. This sequence then has an accumulation point. Then, [Zan69, Lem. 4.2] ensures that the composite map A is closed on $L(x^0)$.

The assumptions of Theorem A are thus fulfilled, and we conclude that $\{T(x^t)\}$ converges to $T(\bar{x})$ for some $\bar{x} \in \Omega$, and any accumulation point of $\{x^t\}$ lies in Ω . (The existence of at least one such point follows from the boundedness of $\{x^t\}$.) The last statement follows from [OrR70, Thm. 14.1.4]. $\square$

Note that it is essential that u is continuous on $L(x^0)$, since otherwise the line search mapping E need not be closed. (In the special case where $u \equiv \delta_X$, the line search is made with respect to f , and the continuity assumption on u may be removed.)

4.2. A truncated algorithm. In this section we provide one example of the many possibilities for constructing realizations of *truncated* CA algorithms. The basis for the algorithm described and validated here is Lemma 3.2(b). (Another truncated CA algorithm, based on Lemma 3.2(c), is presented in section 4.3.) The idea behind a truncated CA method is to reduce the work performed on $[\text{NDP}_\varphi]$ by limiting the number of iterations performed when solving it using a descent algorithm. This strategy introduces a trade-off between the computational effort spent on solving the subproblem and the quality of the search direction obtained. (Examples of such methods are extensions of truncated Newton methods for systems of nonlinear equations, in which a limited number of steps of an iterative scheme is performed on the linear systems.)

The following assumption is made on the algorithm used for solving $[\text{NDP}_\varphi]$.

Assumption 4.1 (properties of a truncated algorithm). Given an $x \in \text{dom } u$, the problem $[\text{NDP}_\varphi]$ is solved, starting from x , with an iterative algorithm where one iteration may be described by a mapping $G : \text{dom } u \times \text{dom } u \mapsto 2^{\text{dom } u} \times \text{dom } u$ having the following properties.

- (1) *Fixed point.* $z \in Y(x) \iff (z, x) \in G(z, x)$, in which case the algorithm terminates.
- (2) *Descent.* If $x \notin \Omega$ and $z \notin Y(x)$, then $T_\varphi(y, x) < T_\varphi(z, x)$ for all y with $(y, x) \in G(z, x)$.
- (3) *Closedness.* The mapping G is upper semicontinuous, closed, and compact valued on $\text{dom } u \times L(x^0)$.
- (4) *Finite termination.* The number of iterations performed is bounded from below by 1 and from above by a positive integer $\bar{k}$.

Remark 4.1. The condition (1) ensures that the fixed point property of the original algorithm is preserved and that the algorithm terminates whenever a solution to $[\text{NDP}_\varphi]$ has been obtained. The condition (2), together with the lower bound in (4) on the number of iterations performed and Lemma 3.2(b), ensures that the resulting approximate solution $\bar{y}$ defines a direction of descent with respect to T . Condition (3) ensures the closedness of the mapping describing the overall algorithm. Whenever $\text{dom } u$ is bounded, it can be replaced by the assumption that the mapping G is closed on $\text{dom } u \times \text{dom } u$. Condition (4), finally, limits the work done in each main iteration of the CA algorithm, and, together with (3), it ensures that any sequence $\{\bar{y}\}$ of approximate solutions generated over a compact set is bounded.

If $\text{dom } u$ is bounded, then the mapping G describing *any* truncated CA algorithm based on a function $\hat{\varphi} : \text{dom } u \times \text{dom } u \mapsto \mathbb{R}$ of the form $\hat{\varphi}(y, x)$, together with *any* closed updating step which guarantees descent (such as Rule E or a fixed relaxation step), satisfies (1)–(3). $\square$

The truncated CA algorithm is obtained from replacing Step 1 of the CA algorithm (see Table 1) by the following.

- 1'. *Search direction generation.* Apply $1 \leq k_t \leq \bar{k}$ iterations of the algorithm described by G , starting from x^t ; that is, let

$$(\bar{y}^t, x^t) \in \underbrace{G \circ \cdots \circ G}_{k_t \text{ times } G}(x^t, x^t).$$

The resulting search direction is $d^t := \bar{y}^t - x^t$.

Note that the value of k_t need not be determined a priori and could instead be regarded as a product of the algorithm and the termination criteria chosen for $[\text{NDP}_\varphi]$.

THEOREM 4.2 (convergence of a truncated algorithm). *Replace Step 1 by Step 1' in the CA algorithm. In Theorem 4.1, replace the assumption that (9) holds for every $x \in L(x^0)$ with Assumption 4.1. Then, the conclusions of Theorem 4.1 hold for the truncated CA algorithm.*

Proof. The proof utilizes Zangwill's theorem and the Spacer Step theorem [Lue84, p. 231]. We begin by identifying the algorithmic mapping A . We assume that $\{x^t\}$ is infinite (otherwise, by Assumption 4.1(1), with $z = x$, the algorithm is terminated at a point in Ω). By Assumption 4.1(4), there must be at least one positive integer, say k , that occurs an infinite number of times in the sequence $\{k_t\}$. We define

$$\bar{Y}(x) := \{ \bar{y} \in \mathbb{R}^n \mid (\bar{y}, x) \in \underbrace{G \circ \cdots \circ G}_{k \text{ times } G}(x, x) \}.$$

The algorithmic mapping is $A := E \circ \bar{D}$, where $\bar{D} := \bar{Y} - \cdot$ and E is the exact line search map.

We next establish that this mapping has the properties desired (boundedness, adaption, closedness). We first note that Assumption 4.1(2), together with Lemma 3.2(b), yields that the mapping $\bar{D}$ has the descent property. From the use of Rule E, it then follows that $T(\bar{x}) < T(x)$ for any $\bar{x} \in E \circ \bar{D}(x)$, and therefore A satisfies the second condition (adaption). The boundedness assumption on the set $L(x^0)$ then implies that the sequence $\{x^t\}$ is bounded, thereby satisfying the first condition of Zangwill (boundedness).

We now establish the third condition (closedness). The mapping G is closed on $\text{dom } u \times L(x^0)$ (Assumption 4.1(3)). By the upper semicontinuity and compactness of G (Assumption 4.1(3) again), the image $\cup_{x \in L(x^0)} G(x, x)$ of G on the compact set $L(x^0)$ is compact [Nik68, Lem. 4.5]. Lemma 12 of [Mey79] then implies that if $\bar{y}^\infty$ is an accumulation point of a sequence $\{\bar{y}^t\}$ given by $(\bar{y}^{t+1}, x) \in G(\bar{y}^t, x)$, then the set $\{y \mid (y, x) \in G \circ G(\bar{y}^\infty, x)\}$ contains an accumulation point of $\{\bar{y}^t\}$. We are then in the position of using Lemma 4.2 of [Zan69] to conclude that the composite mapping $G \circ G$ is closed on $\text{dom } u \times L(x^0)$. This result used repeatedly then establishes that the mapping $\bar{Y}$, and therefore also $\bar{D}$, is closed on $L(x^0)$. Applying Lemma 4.2 of [Zan69] again, we may conclude that the mapping $E \circ \bar{D}$ is closed on $L(x^0)$. Moreover, the boundedness of $L(x^0)$ and the descent property of $E \circ \bar{D}$ ensures that the range of this mapping over $L(x^0)$ is a compact set. We invoke Corollary 4.2.1 of [Zan69] to conclude that the algorithmic mapping A is closed on $L(x^0)$.

The algorithmic mapping identified by extracting the value k from the choices in the sequence $\{k_t\}$ is clearly of the form

$$C(x) = \{ y \in \text{dom } u \mid T(y) \leq T(x) \}, \quad x \in \text{dom } u.$$

We may therefore invoke the Spacer Step theorem [Lue84, p. 231], which guarantees

that the result holds, thanks to the properties of the mapping A . The rest of the proof follows that of Theorem 4.1. $\square$

A natural sufficient condition for the boundedness of $\{\bar{y}^t\}$ is that $\text{dom } u$ is bounded; the special case $u \equiv \delta_X$ for a nonempty, compact, and convex set X is covered in [Pat93b].

4.3. Implementable step length rules.

THEOREM 4.3 (convergence under Rule A'). *Let $\Phi : \text{dom } u \times \text{dom } u \mapsto \mathbb{R}^n$ be a continuous mapping on $\text{dom } u \times \text{dom } u$ of the form $\Phi(y, x)$, maximal and strictly monotone on $\text{dom } u$ with respect to y . Assume that the point x^0 is chosen so that the lower level set $L(x^0)$ is bounded, and assume further that the problem $[\text{GE}_\Phi]$ is well defined in the sense that (8) holds for every $x \in L(x^0)$. Let Rule A' be used. Then, any accumulation point of the sequence $\{x^t\}$ (at least one such point exists) lies in Ω , and (28) holds.*

Proof. Let x^∞ be any accumulation point of the sequence $\{x^t\}$; the existence of such a point follows from the boundedness of $L(x^0)$ and the descent property. Likewise, the sequence $\{y^t\}$ must be bounded by the well-posedness assumption (8); let y^∞ be an arbitrary accumulation point. Hence, $\{d^t\}$ is bounded.

We will next show that $\{d^t\} \rightarrow 0$ must hold; the proof is by contradiction.

By Lemma 3.5(a),

$$T(x^{t+1}) - T(x^t) \leq \alpha \ell_t [\Phi(x^t, x^t) - \Phi(y^t, x^t)]^T d^t < 0, \quad t = 0, 1, \dots$$

If $\{d^t\} \not\rightarrow 0$, then as $\{T(x^{t+1}) - T(x^t)\} \rightarrow 0$ holds, there must be a subsequence $\mathcal{T}$ such that $\{\ell_t\}_{\mathcal{T}} \rightarrow 0$. There must then be an index $\bar{t}$ such that for every $t \geq \bar{t}$ in $\mathcal{T}$, the step length produced by Rule A' is less than the initial trial step; that is,

$$(29) \quad T(x^t + (\ell_t/\beta)d^t) - T(x^t) > \alpha(\ell_t/\beta)[\Phi(x^t, x^t) - \Phi(y^t, x^t)]^T d^t, \quad t \geq \bar{t}, \quad t \in \mathcal{T}.$$

By the convexity of u , we obtain in the limit of $\mathcal{T}$ in the characterization of y^t that $[\Phi(x^\infty, x^\infty) - \Phi(y^\infty, x^\infty)]^T d^\infty \geq [\nabla f(x^\infty) + \xi_u(x^\infty)]^T d^\infty$ holds for any subgradient $\xi_u(x^\infty)$ of u at x^∞ . Hence, $[\Phi(x^\infty, x^\infty) - \Phi(y^\infty, x^\infty)]^T d^\infty \geq T'(x^\infty; d^\infty)$ holds. Dividing the inequality (29) by ℓ_t/β , and taking the limit of $\mathcal{T}$, we obtain, taking into account the above inequality, Assumption 1.1, and [Roc81, Prop. 3G], that $T'(x^\infty; d^\infty) \geq 0$ holds.

A contradiction is now reached in the limit of the inequality (cf. Lemma 3.2(a)) $T'(x^t; d^t) \leq [\nabla f(x^t) + \xi_u(y^t)]^T d^t = [\Phi(x^t, x^t) - \Phi(y^t, x^t)]^T d^t < 0$. We conclude that $\{d^t\} \rightarrow 0$. Using this result in the characterization of y^∞ then yields that $x^\infty \in \Omega$.

The result (28) follows as in Theorem 4.1. $\square$

In order to introduce simpler step length rules, we assume that the mappings Φ^t are *strongly* monotone; we then automatically reach a large freedom of choosing their form; for example, they are not necessarily determined by a continuous mapping on $\text{dom } u \times \text{dom } u$.

THEOREM 4.4 (convergence under strongly monotone mappings Φ^t). *Assume that T is bounded from below on $\mathbb{R}^n$ and ∇f is Lipschitz continuous on $\text{dom } u$. For each t , let Φ^t be strongly monotone and Lipschitz continuous on $\text{dom } u$. Assume further that $m_\Phi := \liminf_{t \rightarrow \infty} \{m_{\Phi^t}\} > 0$ and $M_\Phi := \limsup_{t \rightarrow \infty} \{M_{\Phi^t}\} < +\infty$.*

- (a) *From any starting point $x^0 \in \text{dom } u$ and under Rule E , A' , or R , any accumulation point of $\{x^t\}$ lies in Ω .*
- (b) *If further $\{x^t\}$ is bounded and Ω is nonempty, then under Rules E , A' , or R , (28) holds. Further, under Rule D , at least one accumulation point of $\{x^t\}$ lies in Ω .*

- (c) If further Ω is finite, then under Rule A' or R, $\{x^t\}$ converges to one point in Ω . The same conclusion holds for Rule E whenever $\{\ell_t\}$ is bounded above.

Proof.

(a) We begin by showing that for each of the three step length rules, $\{d^t\} \rightarrow 0$ must hold. From (25) and the use of Rule E follows that, for $\ell \in [0, 1]$,

$$T(x^{t+1}) \leq T(x^t + \ell d^t) \leq T(x^t) + \ell \left(-m_{\Phi^t} + \frac{M_{\nabla f}}{2} \ell \right) \|d^t\|^2, \quad t = 0, 1, \dots$$

Minimizing the right-hand side of this inequality, we obtain

$$\ell_t^* = \min\{1, m_{\Phi^t}/M_{\nabla f}\}.$$

Hence,

$$(30) \quad T(x^{t+1}) - T(x^t) \leq -\rho \|d^t\|^2, \quad t = 0, 1, \dots,$$

with $\rho := \min\{m_{\Phi}^2/(2M_{\nabla f}), M_{\nabla f}/2\}$. The Armijo rule satisfies (30) with $\rho := (2\beta\alpha(1-\alpha)m_{\Phi}^2)/M_{\nabla f}$, so does the relaxation rule, with $\rho := \varepsilon_1\varepsilon_2 M_{\nabla f}/2$.

Since T is lower bounded and, from the above, $\{T(x^t)\}$ is decreasing, $\{T(x^{t+1}) - T(x^t)\} \rightarrow 0$. Thus, by (30), in each of the three rules we obtain that $\{d^t\} \rightarrow 0$.

From (6) and the upper bound on M_{Φ^t} , as $\{d^t\} \rightarrow 0$,

$$(31) \quad \{\nabla f(x^t) + \xi_u(y^t)\} = \{\Phi^t(x^t) - \Phi^t(y^t)\} \rightarrow 0.$$

Assume now that $\{x^t\}$ has an accumulation point, x^∞ , corresponding to a convergent subsequence $\{x^t\}_{t \in \mathcal{T}}$. Since $\{y^t - x^t\} = \{d^t\} \rightarrow 0$, we have that $\{y^t\}_{t \in \mathcal{T}} \rightarrow x^\infty$. From the closedness of ∂u [Roc70a, Thm. 24.4] and the continuity of ∇f , (31) yields that $\{\xi_u(y^t)\}_{t \in \mathcal{T}} \rightarrow \xi_u(x^\infty) \in \partial u(x^\infty)$, and, again appealing to (31), $\nabla f(x^\infty) + \xi_u(x^\infty) = 0$. Thus, $x^\infty \in \Omega$.

(b) The result (28) follows from (a) and [OrR70, Thm. 14.1.4]. Since $\{\ell_t\} \rightarrow 0$ in Rule D, it holds for all $t \geq \bar{t}$ (cf. (25)) that, for some $\sigma > 0$, $T(x^{t+1}) - T(x^t) \leq -\ell_t \sigma \|d^t\|^2$. Hence, eventually $\{T(x^t)\}$ is decreasing, and by the lower boundedness assumption, it converges to a finite value. We therefore must have that $\sigma \sum_{t=\bar{t}}^\infty \ell_t \|d^t\|^2 < \infty$. But $\sum_{t=0}^\infty \ell_t = \infty$ holds, from which it follows that $\liminf_{t \rightarrow \infty} \|d^t\| = 0$ must hold. The corresponding accumulation point of $\{x^t\}$ must then, by the construction of $[\text{GE}_\Phi]$, lie in Ω .

(c) The result follows from the equality $\|x^{t+1} - x^t\| = \ell_t \|d^t\|$, the result of (a) that $\{d^t\} \rightarrow 0$, the upper bound on ℓ_t , and [OrR70, Thm. 14.1.5]. $\square$

The proof for Rule A' shows that convergence is guaranteed for *any* step length rule that results in a larger reduction of T at each iteration than the Armijo rule.

The above result can be improved to require Lipschitz continuity only for Rules R and D [Pat98].

It is to be observed that although the strong monotonicity property of Φ implies that (8) holds for every $x \in \text{dom } u$, neither $\{x^t\}$ nor $\{y^t\}$ are necessarily bounded. Since $\{T(x^t)\}$ converges and $\{d^t\} \rightarrow 0$, however, boundedness of $\{x^t\}$ and $\{y^t\}$ is ensured if the lower boundedness of T is replaced by the stronger condition of *weak coercivity* ($\text{dom } T$ is bounded or $\lim_{\|x\| \rightarrow \infty} T(x) = +\infty$).

The maximal allowed step length in Rule R is bounded by the constant $2m_{\Phi^t}/M_{\nabla f}$. Now, suppose that f may be written as $f := f_1 + f_2$, where f_1 and f_2 are convex with Lipschitz continuous gradients, and that f_1 is strongly convex. For the sake

of this example, we further assume that f_1 is quadratic. To enhance the speed of convergence of the CA algorithm, we consider redefining φ^t as $\varphi^t := \varphi^t + f_1$. (This operation corresponds to applying a cost approximation to the function $[u + f_1] + f_2$, instead of to $u + f$.) Indeed, this operation yields an increased maximal allowed step length since

$$\frac{2m_{(\varphi^t+f_1)}}{M_{\nabla f_2}} = \frac{2(m_{\varphi^t} + m_{f_1})}{M_{\nabla(f-f_1)}} > \frac{2m_{\varphi^t}}{M_{\nabla f}}.$$

It also yields a faster convergence for the other step length rules; for Rule A', observe the change in the linear convergence ratio in Theorem 4.6 below.

From this example, we are lead to conclude that, in order to achieve the best possible rate of convergence, the function $\varphi^t := \varphi^t + f_1$ should be constructed such that $f_2 := f - f_1$ is *not* strongly convex. (The desire to obtain a high rate of convergence must of course be weighed against the computational difficulty of the subproblems.)

Chen and Rockafellar [ChR92] study splitting methods for the problem of finding a zero of the sum of two maximal monotone operators. They argue that all the strong monotonicity inherent in the problem mapping should be kept in the mapping defining the *forward step*; their result corresponds to the above for this special case of CA method.

We conclude this section by combining inexact solutions of $[\text{GE}_{\Phi^t}]$ and inexact line searches into an implementable algorithm.

THEOREM 4.5 (an implementable algorithm). *Let the assumptions of Theorem 4.4 hold. In the CA algorithm, let Step 1 be performed inexactly, in the sense that, for each t , the approximate solution $\bar{y}^t$ satisfies (10), with $\|r^t\| \leq \omega \|\bar{d}^t\|$ for some $\omega < m_{\Phi}$. Then, all the conclusions of Theorem 4.4 hold, with the exception that in step length Rule R, the constant m_{Φ} should be replaced by $m_{\Phi} - \omega$.*

The proof of this result is very similar to that of Theorem 4.4; its basis is a modification of the inequality (25) resulting from (10), in which m_{Φ} is replaced by $m_{\Phi} - \omega$.

A slightly weaker result can be established wherein the condition $\|r^t\| \leq \omega \|\bar{d}^t\|$ is replaced by the requirement that $\{\|r^t\|\} \rightarrow 0$ holds; the argument is, roughly, that if $\|r^t\| \geq m_{\Phi} \|\bar{d}^t\|$ holds for infinitely many t , then $\{\|\bar{d}^t\|\} \rightarrow 0$ must hold; if not, then Lemma 3.2(c) establishes that $\bar{d}^t$ eventually is a direction of descent.

4.4. Linear convergence. For affine mappings Φ , it is possible to show that the CA algorithm converges *Q-linearly* to an element x^* of Ω , that is, that

$$(32) \quad \limsup_{t \rightarrow \infty} \frac{\|x^{t+1} - x^*\|}{\|x^t - x^*\|} =: q < 1.$$

Convergence rests on a nonexpansiveness result for a part of the mapping $x \mapsto Y(x)$.

Let $\Phi(x) := Bx$ for some positive definite matrix $B \in \Re^{n \times n}$. Then, from the characterization $[\text{GE}_{\Phi^t}]$ of the set $Y(x)$, for any $x \in \text{dom } u$,

$$(33) \quad y(x) = [B + \partial u]^{-1}[B - \nabla f](x).$$

We then rewrite (33) as $y(x) := VW(x)$, where

$$V := \left[\frac{1}{m_{\Phi}}(B + \partial u) \right]^{-1}, \quad W := \left[\frac{1}{m_{\Phi}}(B - \nabla f) \right],$$

and m_{Φ} is the modulus of strong monotonicity of Φ .

This rewriting of the characterization of $y(x)$ is done because the operator V can be shown to be *nonexpansive* on $\text{dom } u$; that is,

$$\|V(x) - V(y)\| \leq \|x - y\|, \quad \forall x, y \in \text{dom } u.$$

Indeed, from the monotonicity of ∂u and the positive definiteness of B we have, for all $x, y \in \text{dom } u$, and $\xi_u(x) \in \partial u(x)$, $\xi_u(y) \in \partial u(y)$,

$$\begin{aligned} [\xi_u(x) - \xi_u(y)]^T (x - y) &\geq 0, \\ (x - y)^T B(x - y) &\geq m_\Phi \|x - y\|^2. \end{aligned}$$

Dividing both the above inequalities by m_Φ and adding them, we obtain

$$\left(\left[\frac{1}{m_\Phi} (B + \xi_u) \right] (x) - \left[\frac{1}{m_\Phi} (B + \xi_u) \right] (y) \right)^T (x - y) \geq \|x - y\|^2.$$

But this inequality yields that the operator V is firmly nonexpansive and therefore nonexpansive [Roc76, EcB92].

The importance of this result is that the convergence analysis is simplified since, using Lemma 3.1(a)(1), we immediately have that for any $x^* \in \Omega$,

$$\|y(x) - x^*\| = \|VW(x) - VW(x^*)\| \leq \|W(x) - W(x^*)\|,$$

and it suffices to study the operator norm of W in the vicinity of x^* .

THEOREM 4.6 (linear convergence). *Assume that ∇f is Lipschitz continuous on $\text{dom } u$. Let $\Phi(x) := Bx$, where $B \in \mathbb{R}^{n \times n}$ is positive definite. Assume that, using Rule A' or Rule R , $\{x^t\} \rightarrow x^* \in \Omega$, where f is twice continuously differentiable and $\nabla^2 f(x^*)$ is positive definite. If*

$$(34) \quad \sigma := \frac{M_\Phi}{m_\Phi} \|I - B^{-1} \nabla^2 f(x^*)\| < 1,$$

then $\{x^t\}$ converges Q -linearly with the ratio

$$\begin{aligned} q &\leq \max \left\{ \sigma, 1 - \frac{2\beta(1-\alpha)m_\Phi}{M_{\nabla f}} (1 - \sigma) \right\}, \quad \text{for Rule } A', \text{ and} \\ q &\leq \max \{ \sigma, 1 - \underline{\ell}(1 - \sigma) \} \quad \text{for Rule } R, \end{aligned}$$

where $\underline{\ell} := \inf_t \{\ell_t\} \geq \varepsilon_1$.

Proof. From Lemma 3.1(a)(1) and the nonexpansiveness property of V , we have $\|y^t - x^*\| = \|VW(x^t) - VW(x^*)\| \leq \|W(x^t) - W(x^*) - \nabla W(x^*)(x^t - x^*)\| + \|\nabla W(x^*)(x^t - x^*)\|$, where $\nabla W(x^*) := (1/m_\Phi)(B - \nabla^2 f(x^*))$.

Let $\varepsilon > 0$ be arbitrary. Since $\{x^t\} \rightarrow x^*$, we have for a sufficiently large t that $\|W(x^t) - W(x^*) - \nabla W(x^*)(x^t - x^*)\| \leq \varepsilon \|x^t - x^*\|$. We also have that $\|\nabla W(x^*)\| \leq (M_\Phi/m_\Phi) \|I - B^{-1} \nabla^2 f(x^*)\| =: \sigma$, so that $\|y^t - x^*\| \leq (\sigma + \varepsilon) \|x^t - x^*\|$. Then, $\|x^{t+1} - x^*\| = \|x^t + \ell_t d^t - x^*\| \leq \ell_t \|y^t - x^*\| + (1 - \ell_t) \|x^t - x^*\| \leq [1 - \ell_t(1 - \sigma - \varepsilon)] \|x^t - x^*\|$.

For Rule A' , from (24), we then have, for a large enough t ,

$$(35) \quad \|x^{t+1} - x^*\| \leq q_\varepsilon \|x^t - x^*\|,$$

where $q_\varepsilon := \max\{\sigma + \varepsilon, 1 - 2\beta(1 - \alpha)m_\Phi/M_{\nabla f}(1 - \sigma - \varepsilon)\}$, while, for Rule R , (35) holds for large enough t with $q_\varepsilon := \max\{\sigma + \varepsilon, 1 - \underline{\ell}(1 - \sigma - \varepsilon)\}$. Since ε was arbitrary we have asymptotically that

$$q := \limsup_{t \rightarrow \infty} \frac{\|x^{t+1} - x^*\|}{\|x^t - x^*\|} \leq q_0,$$

where $q_0 := \max\{\sigma, 1 - 2\beta(1 - \alpha)m_\varphi/M_{\nabla f}(1 - \sigma)\}$ in the case of Rule A', and $q_0 := \max\{\sigma, 1 - \ell(1 - \sigma)\}$ for Rule R. If $\sigma < 1$, then $q_0 < 1$, and the theorem is proved. $\square$

If (26) holds, then $\ell_t \equiv 1$ is a valid step length in Rule R, and $q \leq \sigma$ holds. Hence, if $B := \nabla^2 f(x^*)$, then *superlinear* convergence is obtained; that is, $q = 0$ in (32). For Rule A', the same conclusion may be drawn, provided that (23) holds.

The result of Theorem 4.4 suggests a means to choosing a matrix B to obtain linear convergence in practice; choosing Φ^t of the form $x \mapsto B^t x$ for some positive definite matrix B^t which, at least asymptotically, approaches the Hessian $\nabla^2 f(x^t)$, eventually the requirement (34) will be fulfilled. Any matrix B^t generated after this occurrence will be an appropriate choice for the fixed matrix B in the above theorem. Further, the longer this choice is postponed, the closer the value of σ will be to zero; therefore, a convergence rate close to superlinear may be obtained. (In fact, it is possible to extend the above result to incorporate general iteration-dependent mappings Φ^t [Pat98].)

We have not been able to establish linear convergence for Rule E or D, since no lower bound on the respective step length is available.

5. Applications of the convergence analysis. We illustrate in this section further examples of the generality of the framework and demonstrate the strength of the convergence analysis. We demonstrate that the results strengthen those previously obtained for special cases of CA methods and introduce a flexibility in the realizations of existing methods under the same convergence conditions.

5.1. On the linear convergence of CA algorithms. Despite the fact that the linear convergence Theorem 4.6 is here established only for iteration independent and affine cost approximation mappings, it is strong enough to reproduce some well-known linear convergence results in differentiable optimization, as we illustrate below.

COROLLARY 5.1 (linear convergence). *Let the assumptions of Theorem 4.6 hold, and let $B := (1/\gamma)I$, $\gamma > 0$.*

- (a) *Let Rule R be applied, using unit steps. If the method converges, then it converges with a linear rate. Further, the convergence ratio is*

$$(36) \quad q \leq \max\{|1 - \gamma m|, |1 - \gamma M|\},$$

where m and M are the smallest and largest eigenvalues of $\nabla^2 f(x^*)$, respectively.

- (b) *Let further $u \equiv \delta_X$, where X is a nonempty, closed, and convex set in $\mathbb{R}^n$. Then, the algorithm is equivalent to the Goldstein–Levitin–Polyak gradient projection algorithm [Gol64, LeP66]. Moreover, the convergence criterion (26) and the linear convergence ratio (36) coincide with those of this algorithm.*
- (c) *Let further $X := \mathbb{R}^n$. Then, the algorithm is equivalent to the steepest descent method with fixed step length γ [Pol63], with the corresponding convergence criterion and linear convergence ratio.*

Proof.

- (a) We first note that $m_\Phi = M_\Phi = 1/\gamma$. The condition (26) for convergence using unit steps then reduces to $\gamma < 2/M_{\nabla f}$, and since $\|\nabla^2 f(x^*)\| \leq M_{\nabla f}$ and for any symmetric matrix A , $\|I - \gamma A\| < 1$ if $\gamma < 2/\|A\|$, (34) follows.

With $\ell := 1$ we have that $q \leq \sigma := \|I - \gamma \nabla^2 f(x^*)\| = \max\{|1 - m|, |1 - M|\}$, where m and M are, respectively, the smallest and the largest eigenvalues of $\gamma \nabla^2 f(x^*)$.

- (b) Straightforward calculations yield $y^t = P_X(x^t - \gamma \nabla f(x^t))$, where $P_X(z)$ denotes the Euclidean projection of z onto X . Since $\ell_t \equiv 1$, $x^{t+1} = y^t$ follows, and we obtain the Goldstein–Levitin–Polyak gradient projection algorithm. The convergence criterion ($\gamma < 2/M_{\nabla f}$) and linear convergence ratio (36) can be found in [LeP66].
- (c) Follows from (b), with $X = \mathbb{R}^n$. $\square$

5.2. Improvements of existing results. We provide a few examples of previous convergence analyses of special cases of CA algorithms and demonstrate that the analysis in this paper improves upon them.

Application 1. In the CA algorithm, choose the following: $\varphi^t := 0$; each subproblem $[\text{NDP}_\varphi]$ is solved exactly; and the conceptual step length Rule E is used. We then obtain the method of Mine and Fukushima [MiF81]. Comparing their convergence results to that of Theorem 4.1, their convergence conditions include an unnecessary strict convexity assumption on u ; further, they do not recognize that u needs to be continuous. Needless to say, our convergence analysis also allows for a larger flexibility in the realization of this method under the same convergence conditions; for example, using inexact solutions of $[\text{NDP}_\varphi]$ is optional (see Theorem 4.2).

When $u \equiv \delta_X$ for a closed, convex set X in $\mathbb{R}^n$, this algorithm reduces to the Frank–Wolfe method. For this algorithm, the validation of truncated subproblems (Theorem 4.2) is new, and its successful use in several applications (where it is often referred to as an heuristic procedure; see, e.g., [LHB85]) is thus supported in theory. Further, in this special case of $[\text{NDP}]$, the original Armijo step length rule can be used in place of Rule A', and convergence is established without the need for Φ to be strictly monotone (see [Pat93c, Pat98]). As far as the author is aware, previous convergence results for Armijo step length rules within the Frank–Wolfe algorithm have always assumed that ∇f is Lipschitz continuous (e.g., [PsD78, Chap. III.3]), which, however, is unnecessary (cf. Theorem 4.3).

Application 2. In the CA algorithm, choose the following: $\alpha > 0$; $\varphi^t(x) := 1/(2\gamma_t)\|x\|^2$, where $\{\gamma_t\}$ is chosen so that $\inf_t \{\gamma_t\} > 0$ and $\sup_t \{\gamma_t\} < 1/\alpha$ holds; each subproblem $[\text{NDP}_{\varphi^t}]$ is solved exactly; and the Armijo step length rule A' is used. We then obtain the method of Fukushima and Mine [FuM81]. The condition that $\sup_t \{\gamma_t\} < 1/\alpha$ holds is unnecessary in the presence of the standard assumption that the acceptance parameter satisfies $\alpha \in (0, 1)$, as evidenced by Lemma 3.5 and Theorem 4.4. Further, the enforced dependency of α on the possible choices of γ_t in their method has detrimental effects both in terms of allowing less flexibility in the realization of the method, and in terms of the convergence rate, which becomes arbitrarily poor if γ_t is chosen close to $1/\alpha$. Optional realizations of this algorithm include the use of inexact solutions of $[\text{NDP}_{\varphi^t}]$ as well as the use of the step length Rules R and D (cf. Theorems 4.4 and 4.5). Moreover, the larger flexibility with which the function φ can be chosen introduces possibilities to improve the convergence rate of the method, as the next application establishes.

Application 3. Assume that $f \equiv 0$; replacing Rule A' with Rule R using unit steps reduces the above method to the *proximal point algorithm* [Mar70, Roc76]. If u is also differentiable, then by the results of Rockafellar [Roc76] the proximal point algorithm can be made superlinearly convergent by letting $\{\gamma_t\} \rightarrow +\infty$. Superlinear convergence is not possible to obtain in the method of [FuM81] in the nondifferentiable case, since letting $\{\gamma_t\} \rightarrow +\infty$ would destroy the validity of their step length rule. The discussion following Theorem 4.6, however, demonstrates that also in the

nondifferentiable case, a convergence rate as close to superlinear as desired can be obtained in the CA algorithm by introducing second-order information from f into Φ ; we conjecture that the method which is obtained if the matrix B^t is never fixed is actually superlinearly convergent. We note that this result is consistent with the results for gradient related methods in unconstrained differentiable optimization, where superlinear convergence is characterized by the convergence of the search directions toward Newton directions.

Application 4. In the CA algorithm, choose the following: φ^t is strongly convex with a Lipschitz continuous gradient; and Rule R is used. We then obtain the *auxiliary problem principle* of Cohen [Coh78, Coh80] (see, e.g., [Pat93c, LaP94, Pat98] for detailed comparisons). The convergence results of this scheme correspond essentially to that of Theorem 4.4(b) for this special choice of CA algorithm. There are several optional realizations of the algorithm with the same convergence conditions, including the use of step length Rule A', the use of mappings Φ^t that are not gradients, and the use of truncated subproblem solutions.

In the case where $u \equiv \delta_X$ for a closed, convex set in $\mathbb{R}^n$, we may identify several other algorithm frameworks that are included in that defined by the CA method, and the convergence analyses are, in all cases, more limited and weaker than ours; examples of such frameworks include those in [DFL86, LaM90, Tse91, Mig94, ZhM95]. Applications of the convergence analysis of the CA method to differentiable optimization can be found in [Pat93b, Pat93c, Pat98].

5.3. Concluding remarks. The class of CA algorithms is very general; some examples of methods that are included have been mentioned. What we have not mentioned is that the generality present allows one to adapt the algorithm to problem structures (such as choosing Φ to adapt to a separability in u or f); further, the analysis provides convergence results for new methods.

In this paper a convergence analysis was made for the class of CA algorithms when applied to a nondifferentiable optimization program; consequences of the analysis were illustrated, which resulted in several improvements of previous analyses. A convergence analysis is performed for the special case of differentiable, constrained, and unconstrained optimization problems in the author's Ph.D. thesis [Pat93c]; in several instances, in particular concerning the requirements on Φ^t and the convergence rate, the results reached in this paper are improved upon. These, and new developments, will be reported elsewhere in the future [Pat98].

Acknowledgments. The author wishes to acknowledge the many constructive remarks made by the reviewers, the associate editor, and associate professor Torbjörn Larsson.

REFERENCES

- [Arm66] L. ARMIJO, *Minimization of functions having Lipschitz continuous first partial derivatives*, Pacific J. Math., 16 (1966), pp. 1–3.
- [Ber63] C. BERGE, *Topological Spaces*, Oliver & Boyd, Edinburgh, 1963.
- [Bre73] H. BRÉZIS, *Opérateurs Maximaux Monotones et Semi-Groupes de Contractions dans les Espaces de Hilbert*, North-Holland, Amsterdam, 1973.
- [Bro68] F. E. BROWDER, *Nonlinear maximal monotone mappings in Banach spaces*, Math. Ann., 175 (1968), pp. 81–113.
- [ChR92] G. H.-G. CHEN AND R. T. ROCKAFELLAR, *Forward-Backward Splitting Methods in Lagrangian Optimization*, Report, Department of Applied Mathematics, University of Washington, Seattle, WA, 1992.

- [Coh78] G. COHEN, *Optimization by decomposition and coordination: A unified approach*, IEEE Trans. Automat. Control, AC-23 (1978), pp. 222–232.
- [Coh80] G. COHEN, *Auxiliary problem principle and decomposition of optimization problems*, J. Optim. Theory Appl., 32 (1980), pp. 277–305.
- [CoZ84] G. COHEN AND D. L. ZHU, *Decomposition coordination methods in large scale optimization problems: The nondifferentiable case and the use of augmented Lagrangians*, in Advances in Large Scale Systems, Vol. 1, J. B. Cruz, ed., JAI Press, Greenwich, CT, 1984, pp. 203–266.
- [CPS92] R. W. COTTLE, J.-S. PANG, AND R. E. STONE, *The Linear Complementarity Problem*, Academic Press, San Diego, CA, 1992.
- [DeV85] V. F. DEM'YANOV AND L. V. VASIL'EV, *Nondifferentiable Optimization*, Optimization Software, New York, 1985.
- [DFL86] J.-P. DUSSAULT, J. A. FERLAND, AND B. LEMAIRE, *Convex quadratic programming with one constraint and bounded variables*, Math. Programming, 36 (1986), pp. 90–104.
- [EcB92] J. ECKSTEIN AND D. P. BERTSEKAS, *On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators*, Math. Programming, 55 (1992), pp. 293–318.
- [FHN96] M. FUKUSHIMA, M. HADDOU, V. H. NGUYEN, J.-J. STRODIOT, T. SUGIMOTO, AND E. YAMAKAWA, *A parallel descent algorithm for convex programming*, Comput. Optim. Appl., 5 (1996), pp. 5–37.
- [FuM81] M. FUKUSHIMA AND H. MINE, *A generalized proximal point algorithm for certain non-convex minimization problems*, Internat. J. Systems Sci., 12 (1981), pp. 989–1000.
- [Gol64] A. A. GOLDSTEIN, *Convex programming in Hilbert space*, Bull. Amer. Math. Soc., 70 (1964), pp. 709–710.
- [LaM90] T. LARSSON AND A. MIGDALAS, *An algorithm for nonlinear programs over Cartesian product sets*, Optimization, 21 (1990), pp. 535–542.
- [LaP94] T. LARSSON AND M. PATRIKSSON, *A class of gap functions for variational inequalities*, Math. Programming, 64 (1994), pp. 53–79.
- [LHB85] L. J. LEBLANC, R. V. HELGASON, AND D. E. BOYCE, *Improved efficiency of the Frank-Wolfe algorithm for convex network programs*, Transportation Sci., 19 (1985), pp. 445–462.
- [LeP66] E. S. LEVITIN AND B. T. POLYAK, *Constrained minimization methods*, U.S.S.R. Comput. Math. and Math. Phys., 6 (1966), pp. 1–50.
- [Lue84] D. G. LUENBERGER, *Linear and Nonlinear Programming*, 2nd ed., Addison-Wesley, Reading, MA, 1984.
- [Mar70] B. MARTINET, *Regularisation d'inéquations variationnelles par approximations successives*, Revue Française d'Informatique et de Recherche Opérationnelle, R-3 (1970), pp. 154–158.
- [Mey79] G. G. L. MEYER, *Asymptotic properties of sequences iteratively generated by point-to-set maps*, Math. Programming Study, 10 (1979), pp. 115–127.
- [Mig94] A. MIGDALAS, *A regularization of the Frank-Wolfe method and unification of certain nonlinear programming methods*, Math. Programming, 65 (1994), pp. 331–345.
- [MiF81] H. MINE AND M. FUKUSHIMA, *A minimization method for the sum of a convex function and a continuously differentiable function*, J. Optim. Theory Appl., 33 (1981), pp. 9–23.
- [Min62] G. J. MINTY, *Monotone (nonlinear) operators in Hilbert space*, Duke Math. J., 29 (1962), pp. 341–346.
- [Nik68] H. NIKAIIDO, *Convex Structures and Economic Theory*, Academic Press, New York, 1968.
- [OrR70] J. M. ORTEGA AND W. C. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.
- [Pat93a] M. PATRIKSSON, *Partial linearization methods in nonlinear programming*, J. Optim. Theory Appl., 78 (1993), pp. 227–246.
- [Pat93b] M. PATRIKSSON, *A unified description of iterative algorithms for traffic equilibria*, European J. Oper. Res., 71 (1993), pp. 154–176.
- [Pat93c] M. PATRIKSSON, *A unified framework of descent algorithms for nonlinear programs and variational inequalities*, Ph.D. dissertation, Department of Mathematics, Linköping Institute of Technology, Linköping, Sweden, 1993.
- [Pat94a] M. PATRIKSSON, *On the convergence of descent methods for monotone variational inequalities*, Oper. Res. Lett., 16 (1994), pp. 265–269.
- [Pat94b] M. PATRIKSSON, *The Traffic Assignment Problem: Models and Methods*, VSP, Utrecht, 1994.
- [Pat97] M. PATRIKSSON, *Merit functions and descent algorithms for a class of variational inequality problems*, Optimization, 41 (1997), pp. 37–55.

- [Pat98] M. PATRIKSSON, *Nonlinear Programming and Variational Inequalities: A Unified Approach*, Kluwer Academic Publishers, Norwell, MA, to appear.
- [Pol63] B. T. POLYAK, *Gradient methods for the minimisation of functionals*, U.S.S.R. Comput. Math. and Math. Phys., 3 (1963), pp. 864–878.
- [Pol87] B. T. POLYAK, *Introduction to Optimization*, Optimization Software, New York, 1987.
- [PsD78] B. N. PSHENICHNY AND YU. M. DANILIN, *Numerical Methods in Extremal Problems*, Mir Publishers, Moscow, 1978.
- [Rob79] S. M. ROBINSON, *Generalized equations and their solutions, part I: Basic theory*, Math. Programming Study, 10 (1979), pp. 128–141.
- [Roc66] R. T. ROCKAFELLAR, *Characterization of the subdifferentials of convex functions*, Pacific J. Math., 17 (1966), pp. 497–510.
- [Roc69] R. T. ROCKAFELLAR, *Local boundedness of nonlinear, monotone operators*, Michigan Math. J., 16 (1969), pp. 397–407.
- [Roc70a] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
- [Roc70b] R. T. ROCKAFELLAR, *On the maximal monotonicity of subdifferential mappings*, Pacific J. Math., 33 (1970), pp. 209–216.
- [Roc70c] R. T. ROCKAFELLAR, *On the maximality of sums of nonlinear monotone operators*, Trans. Amer. Math. Soc., 149 (1970), pp. 75–88.
- [Roc76] R. T. ROCKAFELLAR, *Monotone operators and the proximal point algorithm*, SIAM J. Control Optim., 14 (1976), pp. 877–898.
- [Roc81] R. T. ROCKAFELLAR, *The Theory of Subgradients and its Applications to Problems of Optimization: Convex and Nonconvex Functions*, Heldermann-Verlag, Berlin, 1981.
- [Sho85] N. Z. SHOR, *Minimization Methods for Non-Differentiable Functions*, Springer-Verlag, Berlin, 1985.
- [StW70] J. STOER AND C. WITZGALL, *Convexity and Optimization in Finite Dimensions I*, Springer-Verlag, Berlin, 1970.
- [Tse91] P. TSENG, *Decomposition algorithm for convex differentiable minimization*, J. Optim. Theory Appl., 70 (1991), pp. 109–135.
- [WoZ96] G. R. WOOD AND B. P. ZHANG, *Estimation of the Lipschitz constant of a function*, J. Global Optim., 8 (1996), pp. 91–103.
- [Zan69] W. I. ZANGWILL, *Nonlinear Programming: A Unified Approach*, Prentice-Hall, Englewood Cliffs, NJ, 1969.
- [Zei90] E. ZEIDLER, *Nonlinear Functional Analysis and Its Applications II/B: Nonlinear Monotone Operators*, Springer-Verlag, New York, 1990.
- [ZhM95] D. ZHU AND P. MARCOTTE, *Coupling the auxiliary problem principle with descent methods of pseudoconvex programming*, European J. Oper. Res., 83 (1995), pp. 670–685.

QUASI-NEWTON BUNDLE-TYPE METHODS FOR NONDIFFERENTIABLE CONVEX OPTIMIZATION*

ROBERT MIFFLIN[†], DEFENG SUN[‡], AND LIQUN QI[‡]

Abstract. In this paper we provide implementable methods for solving nondifferentiable convex optimization problems. A typical method minimizes an approximate Moreau–Yosida regularization using a quasi-Newton technique with inexact function and gradient values which are generated by a finite inner bundle algorithm. For a BFGS bundle-type method global and superlinear convergence results for the outer iteration sequence are obtained.

Key words. Moreau–Yosida regularization, bundle method, quasi-Newton method, superlinear convergence

AMS subject classifications. 65K05, 90C30, 52A41, 90C25

PII. S1052623496303329

1. Introduction. Consider the following minimization problem:

$$(1.1) \quad \min_{x \in \mathbb{R}^n} f(x),$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a possibly nondifferentiable convex function.

Throughout this paper, we use $\|\cdot\|$ to denote the Euclidean vector norm on $\mathbb{R}^n$ or its induced matrix norm on $\mathbb{R}^{n \times n}$. Let M be a symmetric positive definite $n \times n$ matrix. For any $x \in \mathbb{R}^n$ let

$$\|x\|_M^2 = x^T M x.$$

We let F_M be the Moreau–Yosida [19, 27] regularization of f , associated with M , defined by

$$(1.2) \quad F_M(x) = \min_{y \in \mathbb{R}^n} \left\{ f(y) + \frac{1}{2} \|y - x\|_M^2 \right\}.$$

It is well known that F_M is a continuously differentiable convex function defined on $\mathbb{R}^n$ even though f may be nondifferentiable. The derivative of F_M at x is defined by

$$G_M(x) \equiv \nabla F_M(x) = M(x - p(x)) \in \partial f(p(x)),$$

where $p(x)$ is the unique minimizer in (1.2) and ∂f is the subdifferential mapping of f [25]. Here, $p(x)$ is called the proximal point of x . Furthermore, G_M is globally Lipschitz continuous with modulus $\|M\|$, the set of minimizers of (1.1) is exactly the set of minimizers of

$$(1.3) \quad \min_{x \in \mathbb{R}^n} F_M(x),$$

*Received by the editors May 10, 1996; accepted for publication (in revised form) February 24, 1997. This research was supported by the Australian Research Council.

<http://www.siam.org/journals/siopt/8-2/30332.html>

[†]Department of Pure and Applied Mathematics, Washington State University, Pullman, WA 99164-3113 (mifflin@beta.math.wsu.edu). The research of this author was supported by the National Science Foundation grants DMS-9402018 and DMS-9703952.

[‡]School of Mathematics, University of New South Wales, Sydney 2052, Australia (sun@maths.unsw.edu.au, qi@maths.unsw.edu.au). This research was performed while Defeng Sun was on leave from the Institute of Applied Mathematics, Chinese Academy of Sciences, Beijing 100080, People's Republic of China.

and x^* minimizes f if and only if $G_M(x^*) = 0$ and $p(x^*) = x^*$. For additional properties, see [26, 17, 23].

In this paper we use Moreau–Yosida regularization, bundle and quasi-Newton ideas to develop a convergent minimization method for f . We do not assume that the subproblem in (1.2) is solved exactly at each outer iteration nor do we assume f is differentiable at a solution x^* . For a particular BFGS bundle method applied to an approximation of F_M we obtain global and superlinear convergence (of outer iterations) if $\nabla G_M(x^*)$ is positive definite and the directional derivative of G_M is radially Lipschitz continuous at x^* . Related work on this subject appears in [1, 5, 6, 7, 12, 16, 18]. In particular, in [1] global and superlinear convergence results for a BFGS proximal method are given by assuming that f is continuously differentiable and $p(x)$ is computed exactly. In the literature, for example [10], global convergence of particular quasi-Newton methods with inexact gradient values has been discussed. In this paper we approximate F_M in addition to G_M and these two approximations are related.

The plan of this paper is as follows. In section 2 we discuss how a bundle method can be used to satisfy our requirement for approximating $p(x)$. We give the quasi-Newton bundle-type algorithm in section 3 and discuss its global convergence in section 4. In section 5 we discuss global and superlinear convergence of a BFGS bundle-type method. Some concluding remarks are given in section 6.

2. The bundle concept. The bundle idea plays a central role in approximating $F_M(x)$ and $\nabla F_M(x)$ as is developed in [16] and [18], for example. Let $d = y - x$ in (1.2) and minimize over d instead of y . This gives

$$F_M(x) = \min_{d \in \mathbb{R}^n} \left\{ f(x+d) + \frac{1}{2} d^T M d \right\}.$$

Now we consider approximating $f(x+d)$ by a polyhedral function

$$\check{f}(x+d) = \max_{i=1,\dots,j} \{f(u^i) + (g^i)^T(x+d-u^i)\},$$

where the data $(u^i, f(u^i), g^i)$ with $g^i = g(u^i) \in \partial f(u^i)$ constitute a bundle generated sequentially starting from x and $g(x) \in \partial f(x)$ and, possibly, a subset of the previous set used to generate x . Since f is convex, we have

$$(2.1) \quad f(x+d) \geq \check{f}(x+d).$$

If we define a linearization error by letting

$$e(x, u^i) = f(x) - f(u^i) - (g^i)^T(x - u^i),$$

then $\check{f}(x+d)$ can be written as

$$(2.2) \quad \check{f}(x+d) = f(x) + \max_{i=1,\dots,j} \{(g^i)^T d - e(x, u^i)\}.$$

Let

$$(2.3) \quad \begin{aligned} \check{F}_M(x) &= \min_{d \in \mathbb{R}^n} \left\{ \check{f}(x+d) + \frac{1}{2} d^T M d \right\} \\ &= f(x) + \min_{d \in \mathbb{R}^n} \left\{ \max_{i=1,\dots,j} \{(g^i)^T d - e(x, u^i)\} + \frac{1}{2} d^T M d \right\}. \end{aligned}$$

From (2.1) and the definition of $F_M(x)$, we have

$$\check{F}_M(x) \leq F_M(x).$$

So $\check{F}_M(x)$ is an underapproximation of the unknown value $F_M(x)$. Let $d(x)$ solve the minimization problem in (2.3), and let

$$v(x) = \max_{i=1,\dots,j} \{(g^i)^T d(x) - e(x, u^i)\}.$$

Then

$$\check{F}_M(x) = f(x) + v(x) + \frac{1}{2}(d(x))^T M d(x).$$

Let $a(x) = x + d(x)$ be an approximation of $p(x)$, and let

$$\hat{F}_M(x) = f(a(x)) + \frac{1}{2}(d(x))^T M d(x).$$

Since $p(x)$ is the unique minimizer in (1.2), we have

$$F_M(x) \leq \hat{F}_M(x)$$

and equality holds if and only if $a(x) = p(x)$.

Thus, we have the following lemma.

LEMMA 2.1.

- (i) $\check{F}_M(x) \leq F_M(x) \leq \hat{F}_M(x)$.
- (ii) $F_M(x) = \hat{F}_M(x)$ if and only if $a(x) = p(x)$.

This simple lemma plays an important role in the design of our algorithm.

Let

$$(2.4) \quad \varepsilon(x) = \hat{F}_M(x) - \check{F}_M(x).$$

We base our rule for accepting $a(x)$ as an approximation of $p(x)$ on $\varepsilon(x)$ as follows:
Accept if

$$(2.5) \quad \varepsilon(x) \leq \delta(x) \min\{(d(x))^T M d(x), N\},$$

where $\delta(x)$ and N are given positive numbers and $\delta(x)$ is fixed during the bundling process. If (2.5) is not satisfied then we let $u^{j+1} = x + d(x)$ and $g^{j+1} = g(u^{j+1})$, append a new piece $(g^{j+1})^T d - e(x, u^{j+1})$ to (2.2), replace j by $j + 1$, and solve a new subproblem in (2.3) for a new $d(x)$ and a new $\varepsilon(x)$ to be tested in (2.5). If this process, in which $\varepsilon(x)$ and $d(x)$ vary, does not terminate we have the following result.

LEMMA 2.2. *Suppose x does not minimize f . In this subalgorithm, if (2.5) is never satisfied, then*

$$\varepsilon(x) \rightarrow 0.$$

Proof. Following the proof of Proposition 3 in [11] (see also [15] and [8]), we can prove that

$$\check{F}_M(x) \rightarrow F_M(x) \text{ and } \hat{F}_M(x) \rightarrow F_M(x) \text{ as } j \rightarrow \infty.$$

So the result of this lemma follows from (2.4). $\square$

Let

$$\tilde{G}_M(x) = M(x - a(x)) = -Md(x).$$

The following result is a slight extension of Lemma 1 in [12]. For completeness, we give the proof.

LEMMA 2.3.

$$(2.6) \quad \|G_M(x) - \tilde{G}_M(x)\|_{M^{-1}} = \|p(x) - a(x)\|_M \leq \sqrt{2\varepsilon(x)},$$

$$(2.7) \quad \|G_M(x) - \tilde{G}_M(x)\| \leq \sqrt{2\varepsilon(x)\|M\|}.$$

Proof. Define the function $\psi : \mathbb{R}^n \rightarrow \mathbb{R}$ by

$$\psi(z) = f(z) + \frac{1}{2}\|z - x\|_M^2.$$

Since f is convex and $\|z - x\|_M^2$ is a strongly convex quadratic function in z , we have the inequality

$$(2.8) \quad \psi(u) \geq \psi(z) + \omega^T(u - z) + \frac{1}{2}\|u - z\|_M^2 \quad \text{for all } u, z \in \mathbb{R}^n \text{ and all } \omega \in \partial\psi(z).$$

Since $p(x)$ is the argmin in (1.2), $0 \in \partial\psi(p(x))$. Letting $u = a(x)$, $z = p(x)$, and $\omega = 0$ in (2.8) gives

$$\psi(a(x)) \geq \psi(p(x)) + \frac{1}{2}\|a(x) - p(x)\|_M^2;$$

i.e.,

$$\hat{F}_M(x) \geq F_M(x) + \frac{1}{2}\|a(x) - p(x)\|_M^2.$$

Then, from Lemma 2.1 and (2.4), (2.6) holds. Finally, we have

$$\|G_M(x) - \tilde{G}_M(x)\|^2 = \|M(p(x) - a(x))\|^2 \leq \|M\|\|p(x) - a(x)\|_M^2,$$

which when combined with (2.6) implies that (2.7) holds. $\square$

LEMMA 2.4. *If x does not minimize f , then after a finite number of subproblem steps we can find a subproblem solution $d(x)$ such that (2.5) holds.*

Proof. If not, then $j \rightarrow \infty$, so from Lemma 2.2,

$$\varepsilon(x) \rightarrow 0.$$

Then, from Lemma 2.3, $\|\tilde{G}_M(x) - G_M(x)\| \rightarrow 0$. Since x is not an optimal solution, $G_M(x) \neq 0$. So there exists a positive number δ_0 such that $\|\tilde{G}_M(x)\| \geq \delta_0$ when j is sufficiently large. Then, since

$$(2.9) \quad (d(x))^T Md(x) = (\tilde{G}_M(x))^T M^{-1} \tilde{G}_M(x)$$

and (2.5) is not satisfied,

$$\varepsilon(x) > \delta(x) \min \left\{ \frac{\delta_0^2}{\|M\|}, N \right\}$$

for all j sufficiently large. This is a contradiction, because $\varepsilon(x) \rightarrow 0$ when $j \rightarrow \infty$. $\square$

Lemma 2.4 says that a bundle-type algorithm can be used to find a vector $d(x)$ such that (2.5) holds if x is not an optimal solution. This is essential for our algorithm.

A practical stopping test for the overall algorithm is to stop if the subalgorithm generates a solution with

$$(2.10) \quad |v(x)| \leq \text{tol},$$

where tol is a small positive input parameter. See, for example, Theorem 1 in [18].

3. The algorithm. Since F_M is a convex function and G_M is globally Lipschitz continuous, a natural idea is to use a quasi-Newton method, such as the BFGS method, to solve (1.3). The severe practical difficulty with this approach is that we cannot expect to calculate $F_M(x)$ and $G_M(x)$ exactly. To approximate these values appropriately the results of section 2 will be useful.

We use the notation $\varepsilon_k = \varepsilon(x^k)$, $a^k = a(x^k)$, $d^k = d(x^k)$ and so on.

QUASI-NEWTON BUNDLE-TYPE ALGORITHM.

Step 0 (initialization). Let σ , ρ , and N be positive numbers such that $\sigma < 1/2$ and $\rho < 1$. Let $\{\delta_k\}$ be a sequence of positive numbers such that $\sum_{k=0}^{\infty} \delta_k < +\infty$. Let $x^0 \in \mathbb{R}^n$ be an initial solution estimate and B_0 be an $n \times n$ symmetric positive definite matrix. Set $k := 0$ and find d^0 and ε_0 as described in section 2 such that

$$\varepsilon_0 \leq \delta_0 \min\{(d^0)^T M d^0, N\},$$

for example starting the bundle process with $j = 1$ and $u^1 = x^0$.

Step 1 (compute a search direction). If $\|\tilde{G}(x^k)\| = 0$, stop with x^k optimal. Else, compute

$$(3.1) \quad s^k = -B_k^{-1} \tilde{G}_M(x^k).$$

Step 2 (line search). Starting with $m = 0$, let i_k be the smallest nonnegative integer m such that

$$(3.2) \quad \tilde{F}_M(x^k + \rho^m s^k) \leq \hat{F}_M(x^k) + \sigma \rho^m (s^k)^T \tilde{G}_M(x^k),$$

where $\tilde{F}_M(x^k + \rho^m s^k)$ is an underapproximation of F_M at $x^k + \rho^m s^k$ and satisfies

$$(3.3) \quad \begin{aligned} & \hat{F}_M(x^k + \rho^m s^k) - \tilde{F}_M(x^k + \rho^m s^k) \\ & \leq \delta_{k+1} \min\{(d(x^k + \rho^m s^k))^T M d(x^k + \rho^m s^k), N\}. \end{aligned}$$

Set $\tau_k := \rho^{i_k}$ and $x^{k+1} := x^k + \tau_k s^k$.

Step 3 (update the quasi-Newton matrix). Let $\Delta x^k = x^{k+1} - x^k$ and $\Delta y^k = \tilde{G}_M(x^{k+1}) - \tilde{G}_M(x^k)$. If $(\Delta x^k)^T \Delta y^k > 0$, update B_k to B_{k+1} such that B_{k+1} is symmetric and positive definite and satisfies quasi-Newton equation

$$B_{k+1} \Delta x^k = \Delta y^k;$$

otherwise set $B_{k+1} := M$. Set $k := k + 1$ and go to step 1.

At Step 1 if $\|\tilde{G}_M(x^k)\| = 0$ then, from the definition of $\tilde{G}_M(x)$, $\|d(x^k)\| = 0$ and then, from (2.5), $\varepsilon(x^k) = 0$, so (2.7) implies $G_M(x^k) = 0$ and x^k is optimal.

From the discussion given in section 2, if $x^k + \rho^m s^k$ does not minimize f , we can find a vector $d(x^k + \rho^m s^k)$ satisfying (3.3) after a finite number of subproblem steps. So Step 2 proceeds as follows: First compute $d(x^k + \rho^m s^k)$ to satisfy (3.3) and then check

if (3.2) is satisfied. If this is not the case, increase m by 1 and repeat with the new point $x^k + \rho^m s^k$; otherwise set $\tau_k = \rho^{i_k}$ and $x^{k+1} = x^k + \tau_k s^k$ and go to Step 3. If for some candidate nonnegative integer m used in Step 2 $x^k + \rho^m s^k$ is an optimal solution and if tol in (2.10) is zero, then the corresponding bundle subalgorithm execution may not terminate. Throughout the sequel we assume that this situation does not occur by assuming that each subalgorithm execution terminates. The next theorem shows that i_k is well defined at each iteration of the algorithm.

THEOREM 3.1. *If x^k does not minimize f , then there exists a number $\bar{\tau}_k > 0$ such that*

$$(3.4) \quad \check{F}_M(x^k + \tau s^k) \leq \hat{F}_M(x^k) + \sigma \tau (s^k)^T \tilde{G}_M(x^k)$$

holds for all $\tau \in (0, \bar{\tau}_k]$, where $\check{F}_M(x^k + \tau s^k)$, the underapproximation of F_M at $x^k + \tau s^k$, satisfies

$$(3.5) \quad \hat{F}_M(x^k + \tau s^k) - \check{F}_M(x^k + \tau s^k) \leq \delta_{k+1} \min\{(d(x + \tau s^k))^T M d(x + \tau s^k), N\}.$$

Proof. Since x^k does not minimize f , there exists a positive number $\tilde{\tau}_k$ such that for any $\tau \in (0, \tilde{\tau}_k]$, $x^k + \tau s^k$ also does not minimize f . Then by Lemma 2.4 for each $\tau \in (0, \tilde{\tau}_k]$ we can find $d(x^k + \tau s^k)$ such that (3.5) holds. Next we prove this lemma by considering the following two cases.

Case 3.1. $\hat{F}_M(x^k) = F_M(x^k)$. Then from Lemma 2.1, we have

$$a(x^k) = p(x^k).$$

Then,

$$\tilde{G}_M(x^k) = M(x^k - a(x^k)) = M(x^k - p(x^k)) = G_M(x^k),$$

and, since x^k is not a solution, (3.1) implies

$$(s^k)^T G_M(x^k) < 0.$$

Since F_M is continuously differentiable and $\sigma < 1$, there exists a number $\bar{\tau}_k > 0$ ($\bar{\tau}_k \leq \tilde{\tau}_k$) such that for all $\tau \in (0, \bar{\tau}_k]$ we have

$$F_M(x^k + \tau s^k) \leq F_M(x^k) + \sigma \tau (s^k)^T G_M(x^k).$$

This implies that (3.4) holds, because, by Lemma 2.1, $\check{F}_M(x^k + \tau s^k) \leq F_M(x^k + \tau s^k)$.

Case 3.2. $\hat{F}_M(x^k) > F_M(x^k)$. Then when τ is sufficiently small, the right-hand side of (3.4) is greater than $F_M(x^k) + \frac{1}{2}(\hat{F}_M(x^k) - F_M(x^k))$ and, as $\tau \rightarrow 0$, the left-hand side satisfies

$$\check{F}_M(x^k + \tau s^k) \leq F_M(x^k + \tau s^k) \rightarrow F_M(x^k).$$

So there exists a positive number $\bar{\tau}_k$ such that (3.4) is satisfied in this case, too. $\square$

4. Global convergence. Throughout the rest of the paper we assume that the algorithm does not terminate so that $\{x^k\}$ is an infinite sequence.

Since $\sum_{k=0}^{\infty} \delta_k < \infty$, there exists a constant C such that

$$(4.1) \quad \sum_{k=0}^{\infty} \delta_k \leq C.$$

Let

$$D = \{x \in \mathbb{R}^n | F_M(x) \leq F_M(x^0) + 2NC\}.$$

LEMMA 4.1. *For all $k \geq 0$ we have*

$$(4.2) \quad F_M(x^{k+1}) \leq F_M(x^k) + N(\delta_k + \delta_{k+1})$$

and

$$x^k \in D.$$

Proof. By Lemma 2.1 and the algorithm rules, for $k \geq 0$

$$\begin{aligned} F_M(x^{k+1}) &\leq \tilde{F}_M(x^{k+1}) + N\delta_{k+1} \\ &\leq \hat{F}_M(x^k) + \sigma \rho^{i_k}(s^k)^T \tilde{G}_M(x^k) + N\delta_{k+1} \\ &= \hat{F}_M(x^k) - \sigma \rho^{i_k} \tilde{G}_M(x^k)^T B_k^{-1} \tilde{G}_M(x^k) + N\delta_{k+1} \\ &\leq \hat{F}_M(x^k) + N\delta_{k+1} \\ &\leq F_M(x^k) + N(\delta_k + \delta_{k+1}). \end{aligned}$$

Thus, for all $k \geq 0$, (4.2) holds and

$$x^{k+1} \in D.$$

The proof is completed by noting that $x^0 \in D$. $\square$

THEOREM 4.2. *Suppose that f is bounded from below and there exist two positive numbers c_1 and c_2 such that $\|B_k\| \leq c_1$ and $\|B_k^{-1}\| \leq c_2$ for all k . Then any accumulation point of $\{x^k\}$ minimizes f .*

Proof. From Lemma 4.1 we know that $F_M(x^k)$ is bounded from above. On the other hand, since f is assumed to be bounded from below, F_M is also bounded from below. Suppose that $\liminf_{k \rightarrow \infty} F_M(x^k) = F_M^*$. Then, by (4.1), (4.2), and a simple $\epsilon - \delta$ argument, we have $\lim_{k \rightarrow \infty} F_M(x^k) = F_M^*$.

Since $\{\delta_k\} \rightarrow 0$, from Lemma 2.1 and the algorithm rules we have $\{\varepsilon_k\} \rightarrow 0$ and

$$\lim_{k \rightarrow \infty} \tilde{F}_M(x^k) = \lim_{k \rightarrow \infty} \hat{F}_M(x^k) = F_M^*.$$

Thus,

$$\lim_{k \rightarrow \infty} \tau_k(s^k)^T \tilde{G}_M(x^k) = 0,$$

which, from the assumption on $\{B_k\}$, implies that

$$(4.3) \quad \lim_{k \rightarrow \infty} \tau_k \|\tilde{G}_M(x^k)\|^2 = 0.$$

Let $\bar{x}$ be an arbitrary accumulation point of $\{x^k\}$, and let $\{x^k\}_{k \in K}$ be a subsequence converging to $\bar{x}$. By Lemma 2.3

$$(4.4) \quad \lim_{k \rightarrow \infty, k \in K} \tilde{G}_M(x^k) = G_M(\bar{x}).$$

If $\liminf_{k \rightarrow \infty, k \in K} \tau_k > 0$, then from (4.3) and (4.4) we have

$$G_M(\bar{x}) = 0.$$

On the other hand, if $\liminf_{k \rightarrow \infty, k \in K} \tau_k = 0$, then by taking a subsequence, if necessary, we can assume that $\tau_k \rightarrow 0$ for $k \in K$. From the line search stopping rule we have

$$\tilde{F}_M(x^k + \rho^{i_k-1} s^k) > \hat{F}_M(x^k) + \sigma \rho^{i_k-1} (s^k)^T \tilde{G}_M(x^k),$$

where $\rho^{i_k-1} = \tau_k / \rho$. So, by Lemma 2.1, we have

$$F_M(x^k + \rho^{i_k-1} s^k) > F_M(x^k) + \sigma \rho^{i_k-1} (s^k)^T \tilde{G}_M(x^k);$$

i.e.,

$$(4.5) \quad \frac{F_M(x^k + \rho^{i_k-1} s^k) - F_M(x^k)}{\rho^{i_k-1}} > \sigma (s^k)^T \tilde{G}_M(x^k).$$

By (4.4), $\{\tilde{G}_M(x^k)\}_{k \in K}$ is bounded. This, together with the assumption on $\{B_k\}$, implies that $\{s^k\}_{k \in K}$ is bounded. So, by taking a subsequence if necessary, we may assume that

$$\lim_{k \rightarrow \infty, k \in K} s^k = \bar{s}.$$

Since $\{\rho^{i_k-1}\}_{k \in K} \rightarrow 0$, by taking a limit in (4.5) on the subsequence $k \in K$, we obtain

$$(4.6) \quad \bar{s}^T G_M(\bar{x}) \geq \sigma \bar{s}^T G_M(\bar{x}).$$

Also, from the assumption on $\{B_k\}$ we have

$$\bar{s}^T G_M(\bar{x}) \leq -\frac{1}{c_2} \|\bar{s}\|^2,$$

which, combined with (4.6) and the fact that $\sigma < 1$, implies that

$$\bar{s}^T G_M(\bar{x}) = 0 \quad \text{and} \quad \bar{s} = 0.$$

Finally, this combined with the assumption on $\{B_k\}$ implies

$$G_M(\bar{x}) = 0.$$

This completes the proof. $\square$

Based on the results established in [14] and [24], we could discuss local convergence of the proposed quasi-Newton bundle-type methods as in [1] by assuming that the initial point x^0 is sufficiently close to a solution x^* and the initial matrix B_0 is sufficiently close to $\nabla G_M(x^*)$. However, it should be noted that we only use an approximation of the proximal point while in [1] the exact value is used. Here we will not give such a discussion on the local convergence of the proposed methods. In the next section, we will discuss a BFGS bundle-type method for which global and superlinear convergence results are obtained.

5. A BFGS bundle-type method. For given vectors Δx and Δy , the BFGS quasi-Newton update of an $n \times n$ symmetric matrix B is the matrix

$$BFGS(B, \Delta x, \Delta y) := B - \frac{B\Delta x\Delta x^T B}{\Delta x^T B\Delta x} + \frac{\Delta y\Delta y^T}{\Delta x^T \Delta y}$$

(see [9] for instance). If B is positive definite and $\Delta x^T \Delta y > 0$, then the symmetric matrix $B^+ = BFGS(B, \Delta x, \Delta y)$ is also positive definite.

In our BFGS bundle-type method, we will assume that $B_0 = M$ and $\sum_{k=0}^{\infty} \delta_k^{1/3} < \infty$. Let

$$\Delta \bar{y}^k = G_M(x^{k+1}) - G_M(x^k).$$

At each iteration, if the following two conditions are satisfied, we will update B_k to $B_{k+1} = BFGS(B_k, \Delta x^k, \Delta y^k)$; otherwise, we let $B_{k+1} := M$. Given $c_3 \in (0, \infty)$ and $c_4 \in (0, 1)$, these two conditions are

$$(5.1) \quad \|\Delta x^k\|_M(\sqrt{2\varepsilon_k} + \sqrt{2\varepsilon_{k+1}}) \leq c_3(\Delta x^k)^T \Delta y^k$$

and

$$(5.2) \quad 2\|\Delta y^k\|_M(\sqrt{2\varepsilon_k} + \sqrt{2\varepsilon_{k+1}}) \leq \min\{c_4, \delta_k^{1/3} + \delta_{k+1}^{1/3}\}\|\Delta y^k\|^2.$$

In order to employ BFGS results from [3] we need the following results.

LEMMA 5.1. *If conditions (5.1) and (5.2) are satisfied for some $k \geq 0$, then*

$$(5.3) \quad (\Delta x^k)^T \Delta y^k \geq (1/(1+c_3))(\Delta x^k)^T \Delta \bar{y}^k \text{ and } \|\Delta \bar{y}^k\|^2 \geq (1-c_4)\|\Delta y^k\|^2.$$

Proof. From (2.6) in Lemma 2.3,

$$\begin{aligned} (\Delta x^k)^T \Delta y^k &= (\Delta x^k)^T \Delta \bar{y}^k + (\Delta x^k)^T (\Delta y^k - \Delta \bar{y}^k) \\ &\geq (\Delta x^k)^T \Delta \bar{y}^k - \|\Delta x^k\|_M \|\Delta y^k - \Delta \bar{y}^k\|_{M^{-1}} \\ &\geq (\Delta x^k)^T \Delta \bar{y}^k - \|\Delta x^k\|_M (\|\tilde{G}_M(x^k) - G_M(x^k)\|_{M^{-1}} \\ &\quad + \|\tilde{G}_M(x^{k+1}) - G_M(x^{k+1})\|_{M^{-1}}) \\ &\geq (\Delta x^k)^T \Delta \bar{y}^k - \|\Delta x^k\|_M (\sqrt{2\varepsilon_k} + \sqrt{2\varepsilon_{k+1}}) \end{aligned}$$

and

$$\begin{aligned} \|\Delta \bar{y}^k\|^2 &= \|\Delta y^k\|^2 + \|\Delta \bar{y}^k - \Delta y^k\|^2 + 2(\Delta y^k)^T (\Delta \bar{y}^k - \Delta y^k) \\ &\geq \|\Delta y^k\|^2 - 2\|\Delta y^k\|_M \|\Delta \bar{y}^k - \Delta y^k\|_{M^{-1}} \\ &\geq \|\Delta y^k\|^2 - 2\|\Delta y^k\|_M (\sqrt{2\varepsilon_k} + \sqrt{2\varepsilon_{k+1}}). \end{aligned}$$

So, if conditions (5.1) and (5.2) are satisfied, then (5.3) holds. This completes the proof. $\square$

We denote the cosine of the angle between $B_k \Delta x^k$ and Δx^k by

$$\cos \theta_k := \frac{(\Delta x^k)^T B_k \Delta x^k}{\|\Delta x^k\| \|B_k \Delta x^k\|}$$

and the corresponding Rayleigh quotient by

$$q_k := \frac{(\Delta x^k)^T B_k \Delta x^k}{(\Delta x^k)^T \Delta x^k}.$$

Let

$$K := \{0\} \cup \{j | (5.1) \text{ or } (5.2) \text{ does not hold for } k = j - 1\} \equiv \{k_0, k_1, \dots, k_i, \dots\}.$$

This implies that $B_j = M$ for $j \in K$ and B_j is a BFGS update of B_{j-1} for $j \notin K$. Also, let $\lceil \cdot \rceil$ be the roundup operator such that $\lceil t \rceil = i$, when $i - 1 < t \leq i$ for $i \in \{1, 2, \dots\}$.

LEMMA 5.2. *Let $\{B_k\}$ be generated by the BFGS bundle-type algorithm. Suppose that there exist numbers $\alpha_1 > 0$ and $\alpha_2 > 0$ such that*

$$(5.4) \quad (\Delta x^k)^T \Delta y^k \geq \alpha_1 \|\Delta x^k\|^2 \text{ and } (\Delta x^k)^T \Delta y^k \geq \alpha_2 \|\Delta y^k\|^2$$

for all $k \geq 0$. Then for any $w \in (0, 1)$ there exist constants $\beta_1, \beta_2, \beta_3 > 0$ such that, for any k satisfying $k_{i-1} \leq k < k_i - 1$, where $k_{i-1}, k_i \in K$ for some $i \geq 1$, the relations

$$\cos \theta_j \geq \beta_1,$$

$$\beta_2 \leq q_j \leq \beta_3,$$

$$\beta_2 \leq \frac{\|B_j \Delta x^j\|}{\|\Delta x^j\|} \leq \frac{\beta_3}{\beta_1}$$

hold for at least $\lceil w(k - k_{i-1} + 1) \rceil$ values of j satisfying $k_{i-1} \leq j \leq k$.

Proof. For any k satisfying $k_{i-1} \leq k < k_i - 1$, (5.1) and (5.2) hold. Then, from (5.3) and (5.4),

$$(\Delta x^k)^T \Delta y^k \geq \bar{\alpha}_1 \|\Delta x^k\|^2 \text{ and } (\Delta x^k)^T \Delta y^k \geq \bar{\alpha}_2 \|\Delta y^k\|^2$$

hold for all k satisfying $k_{i-1} \leq k < k_i - 1$, where $\bar{\alpha}_1 = \alpha_1/(1 + c_3)$ and $\bar{\alpha}_2 = \alpha_2(1 - c_4)/(1 + c_3)$. Then the results of this lemma follow from the proof of Theorem 2.1 in [3]. $\square$

LEMMA 5.3. *For any nonnegative sequence $\{\delta_k\}_{k \geq 0}$, if $\sum_{k=0}^{\infty} \delta_k < \infty$, then*

$$\prod_{k=0}^{\infty} (1 + \delta_k) < \infty.$$

Proof. This result follows easily from the properties of logarithms. $\square$

LEMMA 5.4. *Relative to the line search there exist positive constants η_1 and η_2 such that either*

$$(5.5) \quad \begin{aligned} \tilde{F}_M(x^k + \tau_k s^k) &\leq \hat{F}_M(x^k) - \eta_1 \frac{((s^k)^T \tilde{G}_M(x^k))^2}{\|s^k\|^2} \\ &\quad - \eta_1/(1 - \sigma) \frac{(s^k)^T (G_M(x^k) - \tilde{G}_M(x^k)) ((s^k)^T \tilde{G}_M(x^k))}{\|s^k\|^2} \end{aligned}$$

or

$$(5.6) \quad \tilde{F}_M(x^k + \tau_k s^k) \leq \hat{F}_M(x^k) + \eta_2 (s^k)^T \tilde{G}_M(x^k).$$

Proof. If (3.2) is satisfied by the integer $m = 0$, then (5.6) holds with $\eta_2 \equiv \sigma$. Suppose that $i_k > 0$, which means that (3.2) fails to be satisfied for $m := i_k - 1$; i.e.,

$$\tilde{F}_M(x^k + (\tau_k/\rho)s^k) > \hat{F}_M(x^k) + \sigma(\tau_k/\rho)(s^k)^T \tilde{G}_M(x^k),$$

which together with Lemma 2.1 implies that

$$F_M(x^k + (\tau_k/\rho)s^k) > F_M(x^k) + \sigma(\tau_k/\rho)(s^k)^T \tilde{G}_M(x^k).$$

Then, using the mean value theorem, we obtain

$$(\tau_k/\rho)(s^k)^T G_M(x^k + \theta(\tau_k/\rho)s^k) > \sigma(\tau_k/\rho)(s^k)^T \tilde{G}_M(x^k),$$

where $\theta \in (0, 1)$. Thus, from the Lipschitz continuity of G_M ,

$$\begin{aligned} & (\tau_k/\rho)(\sigma(s^k)^T \tilde{G}_M(x^k) - (s^k)^T G_M(x^k)) \\ & < (\tau_k/\rho)(s^k)^T (G_M(x^k + \theta(\tau_k/\rho)s^k) - G_M(x^k)) \\ & \leq \|M\|((\tau_k/\rho)\|s^k\|)^2, \end{aligned}$$

which implies that

$$\tau_k > \rho \frac{-((s^k)^T G_M(x^k) - \sigma(s^k)^T \tilde{G}_M(x^k))}{\|M\|\|s^k\|^2}.$$

Substituting this into (3.2) gives

$$\tilde{F}_M(x^k + \tau_k s^k) \leq \hat{F}_M(x^k) - \frac{\rho\sigma}{\|M\|} \frac{((s^k)^T G_M(x^k) - \sigma(s^k)^T \tilde{G}_M(x^k))((s^k)^T \tilde{G}_M(x^k))}{\|s^k\|^2},$$

which gives (5.5) with $\eta_1 = \frac{\rho\sigma(1-\sigma)}{\|M\|}$. $\square$

It was proved in [17] that f is strongly convex on $\mathfrak{R}^n$ if and only if F_M is strongly convex on $\mathfrak{R}^n$. From now on we assume that F_M is strongly convex on D . Then there exists an $\alpha > 0$ such that

$$F_M(z) \geq F_M(x) + G_M(x)^T(z - x) + \frac{\alpha}{2}\|z - x\|^2 \quad \text{for all } x, z \in D,$$

$$(G_M(z) - G_M(x))^T(z - x) \geq \alpha\|z - x\|^2 \quad \text{for all } x, z \in D.$$

This implies that there is a unique minimizer of f in D and that D is bounded. Let $\bar{x}$ be the unique solution. The next result gives R -linear convergence of $\{x^k\}$ to $\bar{x}$.

THEOREM 5.5. *Suppose that F_M is strongly convex on D and $\{B_k\}$ is generated by the BFGS bundle-type method and $x^k \neq \bar{x}$ for all $k \geq 0$. Then $\{x^k\}$ converges to the unique solution $\bar{x}$; moreover,*

$$(5.7) \quad \sum_{k=0}^{\infty} \|x^k - \bar{x}\| < \infty$$

and there are constants $r \in [0, 1)$ and $\bar{C} \in (0, \infty)$ and a positive integer $\bar{k}$ such that for all $k \geq \bar{k}$ we have

$$F_M(x^{k+1}) - F_M(\bar{x}) \leq \bar{C}(r^{1/2})^{k-\bar{k}+1}(F_M(x^{\bar{k}}) - F_M(\bar{x})).$$

Proof. First suppose that K has an infinite number of elements. Since F_M is strongly convex on D and G_M is globally Lipschitz continuous, from [21] or Theorem X.4.2.2 of [13], (5.4) holds for $\alpha_1 = \alpha$ and $\alpha_2 = 1/\|M\|$. So, given $w \in (0, 1)$, from Lemma 5.2 there exist constants $\beta, \beta' > 0$ such that for any k satisfying $k_{i-1} \leq k < k_i - 1$, where $k_{i-1}, k_i \in K$ for some $i \geq 1$, the inequalities

$$(5.8) \quad \cos \theta_j \geq \beta$$

and

$$(5.9) \quad \frac{\|B_j \Delta x^j\|}{\|\Delta x^j\|} \leq \beta'$$

hold for at least $\lceil w(k - k_{i-1} + 1) \rceil$ values of j satisfying $k_{i-1} \leq j \leq k$. Since $B_j = M$ if $j \in K$, we can assume β and β' are such that (5.8) and (5.9) hold for all $j \in K$. We define I to be the set of indices j for which (5.8) and (5.9) hold. Since D is bounded, $\{\|G_M(x^k)\|\}$ is a bounded sequence. From (2.7), (3.3), and (2.9), $\|G_M(x^k) - \tilde{G}_M(x^k)\| = o(\|\tilde{G}_M(x^k)\|)$, so there exists an integer $\bar{k}$ such that for all $k \geq \bar{k}$

$$(5.10) \quad 2\|G_M(x^k)\| \geq \|\tilde{G}_M(x^k)\| \geq \frac{1}{2}\|G_M(x^k)\|$$

and

$$(5.11) \quad \left| -\frac{(s^k)^T(G_M(x^k) - \tilde{G}_M(x^k))((s^k)^T \tilde{G}_M(x^k))}{\|s^k\|^2} \right| \leq \frac{(1-\sigma)\beta^2}{2} \|\tilde{G}_M(x^k)\|^2.$$

Consider an iterate x^j with $j \in I$ and $j \geq \bar{k}$. From Lemma 5.4, (5.8), (5.9), and (5.11), we have that

$$(5.12) \quad \hat{F}_M(x^j) - \check{F}_M(x^j + \tau_j s^j) \geq \eta \|\tilde{G}_M(x^j)\|^2,$$

where $\eta = \frac{1}{2}\eta_1\beta^2$ if (5.5) holds or $\eta = \eta_2\beta/\beta'$ if (5.6) holds. So, from (5.12) and (5.10), for all $j \in I$ and $j \geq \bar{k}$,

$$(5.13) \quad \hat{F}_M(x^j) - \check{F}_M(x^j + \tau_j s^j) \geq \frac{\eta}{4} \|G_M(x^j)\|^2.$$

By strong convexity of F_M and Lemma 4.3 in [1], for all $k \geq 0$,

$$(5.14) \quad \frac{1}{2}\alpha \|x^k - \bar{x}\|^2 \leq F_M(x^k) - F_M(\bar{x}) \leq \frac{2}{\alpha} \|G_M(x^k)\|^2.$$

Then, from Lemma 2.1, (5.13), and the right-side inequality in (5.14), for all $j \in I$ and $j \geq \bar{k}$,

$$(5.15) \quad F_M(x^{j+1}) - F_M(\bar{x}) - \varepsilon_{j+1} \leq \left(1 - \frac{\eta\alpha}{8}\right) (F_M(x^j) - F_M(\bar{x})) + \varepsilon_j.$$

Since $\{\delta_k\} \rightarrow 0$, we can take $\bar{k}$ large enough such that for all $k \geq \bar{k}$

$$(5.16) \quad \frac{16\delta_k \|M^{-1}\| \|M\|^2}{\alpha} \leq \min \left\{ 1, \frac{\eta\alpha}{8} \right\}.$$

By (3.3), (2.9), (5.10), the fact that $G_M(\bar{x}) = 0$, the Lipschitz continuity of G_M with modulus $\|M\|$, and (5.14), for all $k \geq \bar{k}$ we have

$$\begin{aligned}
 \varepsilon_k &\leq \delta_k \|M^{-1}\| \|\tilde{G}_M(x^k)\|^2 \\
 &\leq 4\delta_k \|M^{-1}\| \|G_M(x^k)\|^2 \\
 (5.17) \quad &\leq 4\delta_k \|M^{-1}\| \|M\|^2 \|x^k - \bar{x}\|^2 \\
 &\leq \frac{8\delta_k \|M^{-1}\| \|M\|^2}{\alpha} (F_M(x^k) - F_M(\bar{x})).
 \end{aligned}$$

Then from (5.15)–(5.17), for all $j \in I$ and $j \geq \bar{k}$, we have

$$\begin{aligned}
 (5.18) \quad &\left(1 - 8 \frac{\delta_{j+1} \|M^{-1}\| \|M\|^2}{\alpha}\right) (F_M(x^{j+1}) - F_M(\bar{x})) \\
 &\leq \left(1 - \frac{1}{16} \eta \alpha\right) (F_M(x^j) - F_M(\bar{x})).
 \end{aligned}$$

Since $F_M(x^k) > F_M(\bar{x})$ for all k , (5.18) and (5.16) imply $1 - \frac{1}{16} \eta \alpha > 0$. For $w \in (0, 1)$, let $r = (1 - \frac{1}{16} \eta \alpha)^w$ so that in (5.18)

$$1 - \frac{1}{16} \eta \alpha = r^{1/w}.$$

From (3.1), (3.2), the positivity of σ and τ_k , and the positive definiteness of B_k we have

$$\tilde{F}_M(x^{k+1}) < \hat{F}_M(x^k) \quad \text{for all } k.$$

Combining this with (5.17) and Lemma 2.1 yields for all $j \geq \bar{k}$

$$\begin{aligned}
 &\left(1 - 8 \frac{\delta_{j+1} \|M^{-1}\| \|M\|^2}{\alpha}\right) (F_M(x^{j+1}) - F_M(\bar{x})) \\
 &\leq \left(1 + 8 \frac{\delta_j \|M^{-1}\| \|M\|^2}{\alpha}\right) (F_M(x^j) - F_M(\bar{x})).
 \end{aligned}$$

For $k \geq \bar{k}$, let

$$\delta'_k = \frac{1 + 8 \frac{\delta_k \|M^{-1}\| \|M\|^2}{\alpha}}{1 - 8 \frac{\delta_{k+1} \|M^{-1}\| \|M\|^2}{\alpha}}.$$

For any $k \geq \bar{k}$, there exists $k_{i-1}, k_i \in K$ such that k satisfies $k_{i-1} \leq k < k_i$. If $k_i - k_{i-1} \leq 2$, then, since $k_{i-1} \in K \subseteq I$,

$$\frac{r^{1/w}}{1 - 8 \frac{\delta_{j+1} \|M^{-1}\| \|M\|^2}{\alpha}} < \delta'_j r^{1/w} \quad \text{for all } j \geq \bar{k}$$

and

$$r^{1/w} < r < r^{1/2},$$

we have for k satisfying $k_{i-1} \leq k < k_i$,

$$\begin{aligned}
 F_M(x^{k+1}) - F_M(\bar{x}) &\leq \prod_{j=k_{i-1}}^k \delta'_j r (F_M(x^{k_{i-1}}) - F_M(\bar{x})) \\
 (5.19) \qquad \qquad \qquad &\leq \prod_{j=k_{i-1}}^k \delta'_j (r^{1/2})^{k-k_{i-1}+1} (F_M(x^{k_{i-1}}) - F_M(\bar{x})).
 \end{aligned}$$

On the other hand, if $k_i - k_{i-1} > 2$, then when $k_{i-1} \leq k < k_i - 1$, from Lemma 5.2, there are at least $\lceil w(k - k_{i-1} + 1) \rceil$ elements in $I \cap [k_{i-1}, k]$. So for all k satisfying $k_{i-1} \leq k < k_i - 1$, we have

$$(5.20) \qquad F_M(x^{k+1}) - F_M(\bar{x}) \leq \prod_{j=k_{i-1}}^k \delta'_j r^{k-k_{i-1}+1} (F_M(x^{k_{i-1}}) - F_M(\bar{x})).$$

Therefore,

$$\begin{aligned}
 F_M(x^{k_i}) - F_M(\bar{x}) &\leq \delta'_{k_{i-1}} (F_M(x^{k_{i-1}}) - F_M(\bar{x})) \\
 &\leq \prod_{j=k_{i-1}}^{k_i-1} \delta'_j r^{k_i-k_{i-1}-1} (F_M(x^{k_{i-1}}) - F_M(\bar{x})) \\
 (5.21) \qquad \qquad \qquad &\leq \prod_{j=k_{i-1}}^{k_i-1} \delta'_j (r^{1/2})^{k_i-k_{i-1}+1} (F_M(x^{k_{i-1}}) - F_M(\bar{x})).
 \end{aligned}$$

So, from (5.19)–(5.21), for all k satisfying $k_{i-1} \leq k < k_i$ we have

$$(5.22) \qquad F_M(x^{k+1}) - F_M(\bar{x}) \leq \prod_{j=k_{i-1}}^k \delta'_j (r^{1/2})^{k-k_{i-1}+1} (F_M(x^{k_{i-1}}) - F_M(\bar{x})).$$

Without loss of generality, we can assume that $\bar{k} \in K$. Then, from (5.22), for any $k \geq \bar{k}$ we have

$$F_M(x^{k+1}) - F_M(\bar{x}) \leq \prod_{j=\bar{k}}^k \delta'_j (r^{1/2})^{k-\bar{k}+1} (F_M(x^{\bar{k}}) - F_M(\bar{x})).$$

Since $\sum_{k=0}^{\infty} \delta_k < \infty$, $\sum_{k=\bar{k}}^{\infty} (\delta'_k - 1) < \infty$. So, from Lemma 5.3, there exists a constant $\bar{C} > 0$ such that

$$\prod_{k=\bar{k}}^{\infty} \delta'_k \leq \bar{C}.$$

Then, for all $k \geq \bar{k}$

$$(5.23) \qquad F_M(x^{k+1}) - F_M(\bar{x}) \leq \bar{C} (r^{1/2})^{k-\bar{k}+1} (F_M(x^{\bar{k}}) - F_M(\bar{x})).$$

Using (5.14), (5.23), and the fact that $r < 1$, we have

$$\begin{aligned} \sum_{k=\bar{k}}^{\infty} \|x^k - \bar{x}\| &\leq (2/\alpha)^{1/2} \sum_{k=\bar{k}}^{\infty} (F_M(x^k) - F_M(\bar{x}))^{1/2} \\ &\leq \left[\frac{2\bar{C}(F_M(x^{\bar{k}}) - F_M(\bar{x}))}{\alpha} \right]^{1/2} \sum_{k=\bar{k}}^{\infty} (r^{1/4})^{k-\bar{k}} \\ &< \infty. \end{aligned}$$

If there are only finitely many elements in K , then by following the above proof we can prove the same results as in the case where there are infinitely many elements in K . $\square$

In the next lemma we discuss the boundedness of $\{\|B_k\|\}$ and $\{\|B_k^{-1}\|\}$ which was assumed for convergence in Theorem 4.2.

LEMMA 5.6. *Suppose that F_M is strongly convex on D and $\{B_k\}$ is generated by the BFGS bundle-type method. Furthermore, assume that $\{\Delta x^k\}$ and $\{\Delta \bar{y}^k\}$ are such that for all $k \geq 0$*

$$\frac{\|\Delta \bar{y}^k - H_* \Delta x^k\|}{\|\Delta x^k\|} \leq \varepsilon'_k$$

for some symmetric positive definite matrix H_* and for some sequence $\{\varepsilon'_k\}$ with the property that $\sum_{k=0}^{\infty} \varepsilon'_k < \infty$. Then the sequences $\{\|B_k\|\}$ and $\{\|B_k^{-1}\|\}$ are bounded.

Proof. First suppose K has an infinite number of elements. For k satisfying $k_{i-1} \leq k < k_i - 1$, where $k_{i-1}, k_i \in K$ for some $i \geq 1$, (5.1) and (5.2) hold, and by (5.3)

$$(\Delta x^k)^T \Delta y^k \geq \frac{1}{(1 + c_3)} (\Delta x^k)^T \Delta \bar{y}^k > 0.$$

From Lemma 2.3, (5.2), and (5.3), for all k satisfying $k_{i-1} \leq k < k_i - 1$,

$$\begin{aligned} \frac{\|\Delta y^k - H_* \Delta x^k\|}{\|\Delta x^k\|} &\leq \varepsilon'_k + \frac{\|\Delta y^k - \Delta \bar{y}^k\|}{\|\Delta x^k\|} \\ &\leq \varepsilon'_k + \frac{\sqrt{2\varepsilon_k \|M\|} + \sqrt{2\varepsilon_{k+1} \|M\|}}{\|\Delta x^k\|} \\ &\leq \varepsilon'_k + \frac{\sqrt{\|M\| \|M^{-1}\|}}{2\sqrt{1 - c_4}} (\delta_k^{1/3} + \delta_{k+1}^{1/3}) \frac{\|\Delta \bar{y}^k\|}{\|\Delta x^k\|} \\ &\leq \varepsilon'_k + \frac{\sqrt{\|M\|^3 \|M^{-1}\|}}{2\sqrt{1 - c_4}} (\delta_k^{1/3} + \delta_{k+1}^{1/3}). \end{aligned}$$

Let $\bar{\varepsilon}_k = \varepsilon'_k + \frac{\sqrt{\|M\|^3 \|M^{-1}\|}}{2\sqrt{1 - c_4}} (\delta_k^{1/3} + \delta_{k+1}^{1/3})$. Then for all k satisfying $k_{i-1} \leq k < k_i - 1$, we have

$$\frac{\|\Delta y^k - H_* \Delta x^k\|}{\|\Delta x^k\|} \leq \bar{\varepsilon}_k.$$

From the assumptions that $\sum_{k=0}^{\infty} \varepsilon'_k < \infty$ and $\sum_{k=0}^{\infty} \delta_k^{1/3} < \infty$, it follows that

$$\sum_{k=0}^{\infty} \bar{\varepsilon}_k < \infty.$$

Then, from the proof of Theorem 3.2 in [3], it follows that for all k satisfying $k_{i-1} \leq k < k_i$, $\|B_k\|$ and $\|B_k^{-1}\|$ are bounded with the bound depending on $B_{k_{i-1}}$. Finally, since $B_{k_i} = M$ for all $i \geq 0$, the entire sequences $\{\|B_k\|\}$ and $\{\|B_k^{-1}\|\}$ are bounded. The proof is completed by noting that the case where K has a finite number of elements follows in a similar manner from Theorem 3.2 in [3]. $\square$

LEMMA 5.7. *Suppose that F_M is strongly convex on D and B_k is generated by the BFGS bundle-type method. If the sequences $\{\|B_k\|\}$ and $\{\|B_k^{-1}\|\}$ are bounded, then conditions (5.1) and (5.2) are satisfied for all sufficiently large k , and*

$$(5.24) \quad \|x^k - \bar{x}\| = O(\|\Delta x^k\|), \quad \|x^{k+1} - \bar{x}\| = O(\|\Delta x^k\|).$$

Proof. We first prove that τ_k is bounded away from zero. From the proof of Lemma 5.4, we have

$$\tau_k \geq \min \left\{ 1, \rho \frac{-((s^k)^T G_M(x^k) - \sigma(s^k)^T \tilde{G}_M(x^k))}{\|M\| \|s^k\|^2} \right\}.$$

But, since $\|G_M(x^k) - \tilde{G}_M(x^k)\| = o(\|\tilde{G}_M(x^k)\|)$, $\tilde{G}_M(x^k) = -B_k s^k$, the sequences $\{\|B_k\|\}$ and $\{\|B_k^{-1}\|\}$ are bounded, and $\sigma < 1$, it is not difficult to prove that there exists an integer $\bar{k}$ and a positive constant $\bar{\tau}$ such that for any $k \geq \bar{k}$

$$\tau_k \geq \bar{\tau}.$$

Thus, for all k , τ_k is bounded away from zero.

Since $\Delta x^k = x^{k+1} - x^k = \tau_k s^k = -\tau_k B_k^{-1} \tilde{G}_M(x^k)$, this bound on τ_k and the boundedness of $\{\|B_k\|\}$ and $\{\|B_k^{-1}\|\}$ imply that

$$\|\tilde{G}_M(x^k)\| = O(\|\Delta x^k\|).$$

Then, by the strong convexity of F_M ,

$$\|\tilde{G}_M(x^k)\| \geq \|G_M(x^k)\| - \|G_M(x^k) - \tilde{G}_M(x^k)\| \geq \alpha \|x^k - \bar{x}\| - o(\|\Delta x^k\|);$$

so, the first equality of (5.24) holds. Since $x^{k+1} = x^k + \tau_k s^k$, the first equality of (5.24) and the boundedness of $\{\|B_k^{-1}\|\}$ imply that the second equality of (5.24) holds. From Lemma 2.3, the first inequality in (5.17), and (5.24), we have

$$\begin{aligned} \|\Delta y^k - \Delta \bar{y}^k\|_{M^{-1}} &\leq \sqrt{2\varepsilon_k} + \sqrt{2\varepsilon_{k+1}} \\ &\leq \sqrt{2\delta_k \|M^{-1}\|} \|\tilde{G}_M(x^k)\| + \sqrt{2\delta_{k+1} \|M^{-1}\|} \|\tilde{G}_M(x^{k+1})\| \\ &= \sqrt{2\delta_k \|M^{-1}\|} O(\|G_M(x^k)\|) + \sqrt{2\delta_{k+1} \|M^{-1}\|} O(\|G_M(x^{k+1})\|) \\ &\leq (\sqrt{2\delta_k \|M^{-1}\|} + \sqrt{2\delta_{k+1} \|M^{-1}\|}) O(\|\Delta x^k\|). \end{aligned} \tag{5.25}$$

Therefore, by strong convexity and (5.25),

$$\begin{aligned} (\Delta x^k)^T \Delta y^k &\geq (\Delta x^k)^T \Delta \bar{y}^k - \|\Delta x^k\|_M \|\Delta y^k - \Delta \bar{y}^k\|_{M^{-1}} \\ (5.26) \quad &\geq \alpha \|\Delta x^k\|^2 - o(\|\Delta x^k\|^2) \end{aligned}$$

and

$$\begin{aligned} \|\Delta y^k\| &\geq \|\Delta \bar{y}^k\| - \|\Delta y^k - \Delta \bar{y}^k\| \\ (5.27) \quad &\geq \alpha \|\Delta x^k\| - o(\|\Delta x^k\|). \end{aligned}$$

Then the third inequality in (5.17), (5.24), (5.26), (5.27), and the fact that $\{\delta_k^{1/2}/\delta_k^{1/3}\} \rightarrow 0$ imply that the update conditions (5.1) and (5.2) are satisfied for all sufficiently large k . $\square$

Remark 5.1. A principal contribution of this paper is the update or reset tests (5.1) and (5.2) depending on ε_k and $\delta_k^{1/3}$. From the proof of Lemma 5.7 it can be seen that $\delta_k^{1/3} + \delta_{k+1}^{1/3}$ in (5.2) could be replaced by $\delta_k^\gamma + \delta_{k+1}^\gamma$, where $\gamma < 1/2$ if $\{\delta_k\}$ is chosen such that $\sum_{k=0}^\infty \delta_k^\gamma < \infty$.

THEOREM 5.8. *Suppose that all the assumptions in Lemma 5.6 hold. Then $\{\|B_k\|\}$ and $\{\|B_k^{-1}\|\}$ are bounded, K has finitely many elements, (5.7) holds, and*

$$(5.28) \quad \lim_{k \rightarrow \infty} \frac{\|(B_k - H_*)\Delta x^k\|}{\|\Delta x^k\|} = 0.$$

Proof. The first three results follow from Lemmas 5.6 and 5.7, the definition of K , and Theorem 5.5. So we can assume that there exists an integer $\bar{k}$ such that for any $k \geq \bar{k}$, conditions (5.1) and (5.2) are satisfied. As in the proof of Theorem 5.5, we know that (5.4) holds with $\alpha_1 = \alpha$ and $\alpha_2 = 1/\|M\|$. So, for any $k \geq \bar{k}$, we have

$$(\Delta x^k)^T \Delta y^k \geq \bar{\alpha}_1 \|\Delta x^k\|^2 \text{ and } (\Delta x^k)^T \Delta y^k \geq \bar{\alpha}_2 \|\Delta y^k\|^2,$$

where $\bar{\alpha}_1 = \alpha_1/(1 + c_3)$ and $\bar{\alpha}_2 = \alpha_2(1 - c_4)/(1 + c_3)$. As in the proof of Lemma 5.6, by letting $\bar{\varepsilon}_k = \varepsilon'_k + \frac{\sqrt{\|M\|^3\|M^{-1}\|}}{2\sqrt{1-c_4}}(\delta_k^{1/3} + \delta_{k+1}^{1/3})$, we obtain

$$\sum_{k=0}^\infty \bar{\varepsilon}_k < \infty$$

and for all $k \geq \bar{k}$

$$\frac{\|\Delta y^k - H_* \Delta x^k\|}{\|\Delta x^k\|} \leq \bar{\varepsilon}_k.$$

Then (5.28) follows from the proof of Theorem 3.2 in [3]. $\square$

In order to obtain superlinear convergence for the BFGS bundle-type method, we need further assumptions on G_M . From now on we will assume that G_M is Fréchet differentiable at $\bar{x}$, which, together with assuming that F_M is strongly convex, implies that $\nabla G_M(\bar{x})$ is positive definite and, hence, invertible.

COROLLARY 5.9. *Suppose that F_M is strongly convex on D and G_M is Fréchet differentiable at $\bar{x}$. If there exists a constant $L > 0$ such that*

$$(5.29) \quad \frac{\|\Delta \bar{y}^k - \nabla G_M(\bar{x}) \Delta x^k\|}{\|\Delta x^k\|} \leq L \max\{\|x^{k+1} - \bar{x}\|, \|x^k - \bar{x}\|\},$$

then the sequence $\{x^k\}$ generated by the BFGS bundle-type method satisfies

$$(5.30) \quad \lim_{k \rightarrow \infty} \frac{\|(B_k - \nabla G_M(\bar{x}))\Delta x^k\|}{\|\Delta x^k\|} = 0.$$

Moreover, the sequences $\{\|B_k\|\}$ and $\{\|B_k^{-1}\|\}$ are bounded.

Proof. By using Theorems 5.5 and 5.8, (5.29), and Lemma 5.6 with $H_* = \nabla G_M(\bar{x})$ we obtain the results. $\square$

Recall that a Lipschitz continuous function $H : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$ is said to be directionally differentiable of degree 2 at x if

$$H(x + d) - H(x) - H'(x; d) = O(\|d\|^2),$$

where $H'(x; d)$ is the directional derivative of H at x in the direction d [22]. If $\{\|B_k\|\}$ and $\{\|B_k^{-1}\|\}$ are bounded, then (5.29) is satisfied if G_M is differentiable and directionally differentiable of degree 2 at $\bar{x}$. In fact, in this case, from Proposition 2.2 in [24], there exists a constant L_1 such that

$$(5.31) \quad \|\Delta \bar{g}^k - \nabla G_M(\bar{x})\Delta x^k\| \leq L_1 \max\{\|x^{k+1} - \bar{x}\|^2, \|x^k - \bar{x}\|^2\}.$$

On the other hand, from (5.24), there exists a constant L_2 such that

$$\max\{\|x^{k+1} - \bar{x}\|, \|x^k - \bar{x}\|\} \leq L_2 \|\Delta x^k\|,$$

which, together with (5.31), implies that (5.29) holds with $L := L_1 L_2$.

If we do not wish to assume that $\{\|B_k\|\}$ and $\{\|B_k^{-1}\|\}$ are bounded, we may use Corollary 5.9 to obtain such boundedness by assuming that $G'_M(x; \cdot)$ is radially Lipschitz continuous at $\bar{x}$; i.e., the directional derivative of G_M exists on a neighborhood of $\bar{x}$ and there exists a constant $L > 0$ such that

$$\sup_{\|d\|=1} \|G'_M(x; d) - G'_M(\bar{x}; d)\| \leq L\|x - \bar{x}\|$$

for all x in that neighborhood of $\bar{x}$. From Lemma 2.2 in Pang [20], this strong condition implies that (5.29) is satisfied. Also, from results in [20] and [24], this condition implies that G_M is strongly differentiable and directionally differentiable of degree 2 at $\bar{x}$.

LEMMA 5.10. *Suppose that all the assumptions in Corollary 5.9 hold. Then*

$$(5.32) \quad \|x^k + s^k - \bar{x}\| = o(\|x^k - \bar{x}\|).$$

Proof. Since Δx^k is a positive multiple of $s^k = -B_k^{-1}\tilde{G}_M(x^k)$, (5.30) implies that

$$\lim_{k \rightarrow \infty} \frac{\|(B_k - \nabla G_M(\bar{x}))s^k\|}{\|s^k\|} = 0$$

and

$$-\tilde{G}_M(x^k) - \nabla G_M(\bar{x})s^k = o(\|s^k\|).$$

So,

$$\begin{aligned} \nabla G_M(\bar{x})s^k &= -\tilde{G}_M(x^k) + o(\|s^k\|) \\ &= -G_M(x^k) + o(\|G_M(x^k)\|) + o(\|s^k\|) \\ &= O(\|x^k - \bar{x}\|) + o(\|s^k\|), \end{aligned}$$

which together with the invertibility of $\nabla G_M(\bar{x})$ implies that

$$(5.33) \quad \|s^k\| = O(\|x^k - \bar{x}\|).$$

On the other hand,

$$\begin{aligned} (B_k - \nabla G_M(\bar{x}))s^k &= -\tilde{G}_M(x^k) - \nabla G_M(\bar{x})s^k \\ &= -G_M(x^k) + o(\|G_M(x^k)\|) - \nabla G_M(\bar{x})s^k \\ &= -\nabla G_M(\bar{x})(x^k - \bar{x}) - \nabla G_M(\bar{x})s^k + o(\|x^k - \bar{x}\|) \\ (5.34) \quad &= -\nabla G_M(\bar{x})(x^k + s^k - \bar{x}) + o(\|x^k - \bar{x}\|). \end{aligned}$$

From (5.34) and (5.33),

$$\begin{aligned} \frac{\|\nabla G_M(\bar{x})(x^k + s^k - \bar{x})\|}{\|x^k - \bar{x}\|} &= o(1) + \frac{\|(B_k - \nabla G_M(\bar{x}))s^k\|}{\|s^k\|} \frac{\|s^k\|}{\|x^k - \bar{x}\|} \\ &= o(1), \end{aligned}$$

which together with the invertibility of $\nabla G_M(\bar{x})$ implies (5.32). $\square$

LEMMA 5.11. *Suppose that all the assumptions in Corollary 5.9 hold. Then*

$$F_M(x^k + s^k) \leq F_M(x^k) + \sigma(s^k)^T \tilde{G}_M(x^k)$$

for all sufficiently large k .

Proof. From the differentiability of G_M and the fact that $G_M(\bar{x}) = 0$, we have

$$F_M(x) = F_M(\bar{x}) + \frac{1}{2}(x - \bar{x})^T \nabla G_M(\bar{x})(x - \bar{x}) + o(\|x - \bar{x}\|^2).$$

From Lemma 5.10, $\|x^k + s^k - \bar{x}\| = o(\|x^k - \bar{x}\|)$, so

$$\|x^k - \bar{x}\| = \|s^k\| + o(\|s^k\|).$$

Therefore, from Lemma 5.10 and the boundedness of $\{B_k\}$,

$$\begin{aligned} &F_M(x^k + s^k) - F_M(x^k) - \sigma(s^k)^T \tilde{G}_M(x^k) \\ &= -\frac{1}{2}(x^k - \bar{x})^T \nabla G_M(\bar{x})(x^k - \bar{x}) + o(\|x^k - \bar{x}\|^2) - \sigma(s^k)^T \tilde{G}_M(x^k) \\ &= -\frac{1}{2}(s^k)^T \nabla G_M(\bar{x})s^k + o(\|s^k\|^2) + \sigma(s^k)^T B_k s^k \\ &= -\sigma(s^k)^T (\nabla G_M(\bar{x}) - B_k)s^k + \left(\sigma - \frac{1}{2}\right)(s^k)^T \nabla G_M(\bar{x})s^k + o(\|s^k\|^2) \\ &= \left(\sigma - \frac{1}{2}\right)(s^k)^T \nabla G_M(\bar{x})s^k + o(\|s^k\|^2), \end{aligned}$$

which, together with the positive definiteness of $\nabla G_M(\bar{x})$ and the algorithm assumption that $\sigma < 1/2$, implies that for all sufficiently large k ,

$$F_M(x^k + s^k) - F_M(x^k) - \sigma(s^k)^T \tilde{G}_M(x^k) < 0.$$

This completes the proof of this lemma. $\square$

Now we have all the necessary material to give the superlinear convergence result.

THEOREM 5.12. *Suppose that F_M is strongly convex on D , G_M is Fréchet differentiable at $\bar{x}$, and there exists a constant $L > 0$ such that (5.29) holds. Then the sequence $\{x^k\}$ generated by the BFGS bundle-type method converges to $\bar{x}$ Q -superlinearly.*

Proof. From Lemmas 5.11 and 2.1 and line search criterion (3.2), for all sufficiently large k , we have

$$x^{k+1} = x^k + s^k.$$

Then the Q -superlinear convergence of $\{x^k\}$ follows from Lemma 5.10. $\square$

6. Conclusions. This paper presents a globally and superlinearly convergent BFGS bundle-type method for the case where the Moreau–Yosida regularization function F_M and its gradient G_M are computed only approximately. It does not require the original objective to be differentiable at the solution. To accomplish this we employ a bundle method to implement $\varepsilon_k = \hat{F}_M(x^k) - \tilde{F}_M(x^k) = o(\|G_M(x^k)\|^2)$, which is an essential condition for superlinear convergence of an approximate Newton method applied to this type of problem [12]. Because of this requirement the subproblems may increase in difficulty as k increases. To try to alleviate this potential difficulty it may be beneficial to consider space decomposition as in [18] and to vary M in such a way that the subproblems are solved mainly in the subspace where the cutting-plane aspect of bundling is efficient. Also, if the variation in M and space decomposition are done properly, it may be possible to weaken the rate of convergence assumption to assuming that some regularization of f is strongly convex on a proper subset of $\mathbb{R}^n$ when f is not differentiable at the solution.

In [7], Chen and Fukushima provide a globally and linearly convergent proximal quasi-Newton method and discuss local superlinear convergence conditions. Here we focus our attention on giving superlinear convergence conditions for a BFGS bundle-type method. It may be possible to generalize our results to an important subclass of the Broyden class of quasi-Newton methods by using the results in [4, 2] corresponding to some positive and negative values of the class parameter.

Acknowledgment. The authors would like to thank two referees for their helpful comments.

REFERENCES

- [1] J. F. BONNANS, J. CH. GILBERT, C. LEMARÉCHAL, AND C. A. SAGASTIZÁBAL, *A family of variable metric proximal methods*, Math. Programming, 68 (1995), pp. 15–47.
- [2] R. H. BYRD, D. C. LIU, AND J. NOCEDAL, *On the behavior of Broyden’s class of quasi-Newton methods*, SIAM J. Optim., 2 (1992), pp. 533–557.
- [3] R. H. BYRD AND J. NOCEDAL, *A tool for the analysis of quasi-Newton methods with application to unconstrained minimization*, SIAM J. Numer. Anal., 26 (1989), pp. 727–739.
- [4] R. H. BYRD, J. NOCEDAL, AND Y. YUAN, *Global convergence of a class of quasi-Newton methods on convex problems*, SIAM J. Numer. Anal., 24 (1987), pp. 1171–1189.
- [5] J. V. BURKE AND M. QIAN, *A variable metric proximal point algorithm for monotone operators*, SIAM J. Optim., to appear.
- [6] J. V. BURKE AND M. QIAN, *On the superlinear convergence of the variable metric proximal point algorithm using Broyden and BFGS matrix secant updating*, Math. Programming, 1998, submitted.
- [7] X. CHEN AND M. FUKUSHIMA, *Proximal Quasi-Newton Methods for Nondifferentiable Convex Optimization*, Applied Mathematics report 95/32, School of Mathematics, The University of New South Wales, Sydney, Australia, 1995.
- [8] R. CORREA AND C. LEMARÉCHAL, *Convergence of some algorithms for convex optimization*, Math. Programming, 62 (1993), pp. 261–275.

- [9] J. E. DENNIS AND J. J. MORÉ, *Quasi-Newton methods, motivation and theory*, SIAM Rev., 19 (1977), pp. 46–89.
- [10] U. FELGENHAUER, *On the stable global convergence of particular quasi-Newton methods*, Optimization, 26 (1992), pp. 97–113.
- [11] M. FUKUSHIMA, *A descent algorithm for nonsmooth convex optimization*, Math. Programming, 30 (1984), pp. 163–175.
- [12] M. FUKUSHIMA AND L. QI, *A globally and superlinearly convergent algorithm for nonsmooth convex minimization*, SIAM J. Optim., 6 (1996), pp. 1106–1120.
- [13] J. B. HIRIART-URRUTY AND C. LEMARÉCHAL, *Convex Analysis and Minimization Algorithms*, vols. 1 and 2, Springer-Verlag, Berlin, Heidelberg, New York, 1993.
- [14] C. -M. IP AND J. KYPARISIS, *Local convergence of quasi-Newton methods for B-differentiable equations*, Math. Programming, 56 (1992), pp. 71–89.
- [15] K. C. KIWIEL, *Proximal control in bundle methods for convex nondifferentiable minimization*, Math. Programming, 46 (1990), pp. 105–122.
- [16] C. LEMARÉCHAL AND C. SAGASTIZÁBAL, *An approach to variable metric methods*, in *Systems Modelling and Optimization*, in Lecture Notes in Control and Inform. Sci. 197, J. Henry and J. -P. Yvon, eds., Springer-Verlag, Berlin, 1994.
- [17] C. LEMARÉCHAL AND C. SAGASTIZÁBAL, *Practical aspects of the Moreau-Yosida regularization I: Theoretical preliminaries*, SIAM J. Optim., 7 (1997), pp. 367–385.
- [18] R. MIFFLIN, *A quasi-second-order proximal bundle algorithm*, Math. Programming, 73 (1996), pp. 51–72.
- [19] J. J. MOREAU, *Proximité et dualité dans un espace hilbertien*, Bull. Soc. Math. France, 93 (1965), pp. 273–299.
- [20] J. -S. PANG, *Newton's method for B-differentiable equations*, Math. Oper. Res., 15 (1990), pp. 311–341.
- [21] M. J. D. POWELL, *Some global convergence properties of a variable metric algorithm for minimization without exact line searches*, in *Non-Linear Programming*, R. W. Cottle and C. E. Lemke, eds., SIAM-AMS Proceedings 9, AMS, Providence, RI, 1976.
- [22] L. QI, *Convergence analysis of some algorithms for solving nonsmooth equations*, Math. Oper. Res., 18 (1993), pp. 227–244.
- [23] L. QI, *Second-order analysis of the Moreau-Yosida regularization of a convex function*, Applied Mathematics report 94/20, revised version, School of Mathematics, The University of New South Wales, Sydney, Australia, 1995.
- [24] L. QI, *On superlinear convergence of quasi-Newton methods for nonsmooth equations*, Oper. Res. Lett., 20 (1997), pp. 223–228.
- [25] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
- [26] R. T. ROCKAFELLAR, *Maximal monotone relations and the second derivatives of nonsmooth functions*, Ann. Inst. H. Poincaré Anal. Non Linéaire, 2 (1985), pp. 167–186.
- [27] K. YOSIDA, *Functional Analysis*, Springer-Verlag, Berlin, 1964.

NUMERICAL EXPERIENCE WITH LOWER BOUNDS FOR MIQP BRANCH-AND-BOUND*

ROGER FLETCHER[†] AND SVEN LEYFFER[†]

Abstract. The solution of convex mixed-integer quadratic programming (MIQP) problems with a general branch-and-bound framework is considered. It is shown how lower bounds can be computed efficiently during the branch-and-bound process. Improved lower bounds such as the ones derived in this paper can reduce the number of quadratic programming (QP) problems that have to be solved. The branch-and-bound approach is also shown to be superior to other approaches in solving MIQP problems. Numerical experience is presented which supports these conclusions.

Key words. integer programming, mixed integer quadratic programming, branch-and-bound

AMS subject classifications. 90C10, 90C11, 90C20

PII. S1052623494268455

1. Introduction. One of the most successful methods for solving mixed-integer nonlinear problems is branch-and-bound. Land and Doig [17] first introduced a branch-and-bound algorithm for the travelling salesman problem. Dakin [4] introduced the now common branching dichotomy and was the first to realize that it is possible to solve mixed-integer nonlinear programming problems by branch-and-bound. This paper presents a general framework algorithm for convex mixed-integer quadratic programming (MIQP) branch-and-bound and shows, in particular, how lower bounds can be computed efficiently. We use the abbreviations MI (mixed integer), LP (Linear programming), QP (quadratic programming), and NLP (nonlinear programming).

The introduction of MIQP master programs into outer approximation [8] has motivated us to study the solution of MIQP problems in detail. However, MIQP problems are not only interesting as master problems in the aforementioned algorithm but also in their own right, having various applications such as index tracking for passive portfolio management [16], the optimal sizing and siting of substations in a network routing problem, and so on.

Several authors suggest branching rules in the context of mixed-integer linear programming problems (e.g., [1], [2], [3], [20]), and some of these rules have been adopted to the mixed-integer nonlinear case (cf. [13]). However, despite their potential importance no attempt has been made to derive lower bounds for the MIQP branch-and-bound algorithm. This paper suggests a procedure which provides lower bounds that can be used in branch-and-bound. Our numerical experience suggests that improved lower bounds can considerably reduce the number of QP problems solved.

Branch-and-bound is not the only approach to solving MIQP problems. In a recent paper, Lazimy [18] shows how generalized Benders decomposition can be employed to solve MIQP problems via a finite sequence of QP subproblems and MILP master programs. Flippo and Rinnoy Kan [9] improve Lazimy's approach so that it becomes applicable to a wider class of MIQP problems. Alternatively, outer approx-

*Received by the editors May 25, 1994; accepted for publication (in revised form) February 28, 1997. This work was supported by SERC grant SERC GR/F 07972.

<http://www.siam.org/journals/siopt/8-2/26845.html>

[†]University of Dundee, Department of Mathematics, Dundee, DD1 4HN, Scotland, UK (fletcher@mcs.dundee.ac.uk, sleyffer@mcs.dundee.ac.uk).

imation (Duran and Grossmann [5]) or LP/QP based branch-and-bound (Quesada and Grossmann [21]) can readily be adapted [19] to solve MIQP problems. However, all these approaches are essentially based upon linearizations and consequently ignore higher order terms. Our own experience, presented in section 5, suggests that they are inferior to branch-and-bound. A review of different approaches to MIQP problems is also given by Volkovich, Roshchin, and Sergienko [23].

The present paper is organized in six sections. In section 2 the notation and terminology required to describe our MIQP branch-and-bound algorithm is introduced. In section 3 the general method to derive improved lower bounds is presented, and it is shown how these lower bounds can be obtained efficiently. A small example is presented in section 4 together with some numerical results for the branch-and-bound solver. In section 5 results for the other MIQP solvers are presented and compared to the branch-and-bound solver. The conclusions are summarized in section 6.

2. Notation and terminology. Branch-and-bound is a general framework for solving integer and combinatorial problems. The combinatorial part of the problem (determining the optimal integer assignment) is solved by a tree search in which QP relaxations of the MIQP problem are solved and noninteger QP solutions are eliminated by adding simple bounds (branching). By using lower and upper bounds on the optimal objective value it is possible to limit the tree search, thus avoiding complete enumeration.

The branch-and-bound methodology for solving MIQP problems is very similar to Beale's [1] paper on MILP branch-and-bound. The particular problem considered here is

$$\mathcal{P} \begin{cases} \min_x & f(x) = \frac{1}{2}x^T Gx + g^T x \\ \text{subject to} & A^T x \geq b, \\ & x \in X, \ x_i \text{ integer } \forall i \in I, \end{cases}$$

where f is a convex function and the set X represents simple bounds on the variables. It is assumed that the feasible region is bounded, which can be achieved for instance by ensuring that X is bounded. There is no difficulty in extending the methods of this paper to solve problems where some of the "integer" variables are restricted to take a set of discrete values other than just the integers.

In order to describe a branch-and-bound algorithm it is necessary to introduce some notation and terminology. Let $\mathcal{P}'$ denote the problem obtained from $\mathcal{P}$ by relaxing all integer restrictions. Problem $\mathcal{P}'$ is then an ordinary QP problem. The branch-and-bound algorithm starts by solving $\mathcal{P}'$, giving x' as its solution. If x' satisfies all integer restrictions it is said to be an *integer feasible solution* and in this case also solves $\mathcal{P}$ and the algorithm stops. Otherwise there exists a variable x'_j , $j \in I$ which is not integer (and is said to be *fractional*). Branch-and-bound then proceeds by *branching* on a fractional variable, x'_j say. This is done by introducing two new subproblems obtained from $\mathcal{P}'$ by adding the simple bound $x_j \geq [x'_j + 1]$ to one and $x_j \leq [x'_j]$ to the other, where $[a]$ denotes the largest integer not greater than a . The solution of $\mathcal{P}$ is contained in the feasible region of one of the two new subproblems, and the process can be repeated.

The branch-and-bound algorithm continues to solve and generate new subproblems, performing a tree search where the nodes of the tree correspond to QP subproblems. The nodes obtained by branching are called *child nodes* and the node from which they are generated is called the *parent node*. A node which has been fully explored is referred to as being *fathomed*. A *pending node* is a node which has

been generated by branching but has not yet been solved. The branch-and-bound algorithm searches the tree until no pending nodes remain.

It is not always necessary to search the complete tree, and the success of branch-and-bound is partly due to the fact that whole subtrees can be excluded from further consideration while fathoming their respective root node. A node has been fathomed if it is recognized as infeasible or produces an integer feasible solution. In the latter case the optimal value of the node also provides an *upper bound* on the solution of $\mathcal{P}$. This upper bound is then used to fathom nodes whose optimal value or lower bound is greater than or equal to the current upper bound. The last point indicates that the existence of easy-to-compute lower bounds on the value of a node can reduce the number of problems that need to be solved. For more details we refer the reader to [10, Chapter 10], [22, page 171], and [13].

In the remainder of this section we briefly comment on a number of implementation details that are left open by a general description of the branch-and-bound algorithm. The first question concerns the choice of QP solver. Using a dual method to solve the subproblems offers the most straightforward way to exploit the structure which the branching introduces into the problem. Branching makes the solution of the parent problem infeasible, but a dual feasible point is readily available from the dual solution of the parent problem. A dual active set method could therefore take immediate advantage of a feasible starting point. Moreover, since a dual active set method is an ascend method it can make use of an upper bound as a cut-off value terminating a QP solve prematurely. The present implementation of an MIQP branch-and-bound algorithm uses a primal active set method with six different degrees of warm starts to solve the QP subproblems. The warm starts enable the QP solver to recycle information from the solve of the parent problem. This solver has the additional advantage that it implements a technique to resolve degeneracy and thus gives guaranteed termination even in the presence of round-off errors [7]. This guarantee contributes to the robustness of the MIQP code.

Another question is that concerning the choice of the pending node to be solved next. This is resolved in favor of a depth first search of the tree with the additional feature that backtracking is done to the most promising node. The aim is to quickly find an integer feasible solution so that the resulting upper bound can be used to fathom nodes whose lower bounds exceed the new upper bound.

Finally, the important question regarding the choice of the branching variable is discussed. Many authors suggest a branching rule based on the importance of the integer variables (see [1] for MILP and [13] for MINLP). This rule exploits user-defined priorities which are assigned to the integer variables and branches on the most important variable first.

For variables of equal priority tie-breaking rules are required and these are discussed next. A very simple branching rule which has proved successful both for MINLP branch-and-bound [13] and the travelling salesman problem [20] is based on the amount by which the variables violate the integer restrictions. This strategy selects the integer variable which is furthest from its nearest integer value and is referred to as *maximal fractional part branching* by Breu and Burdet [3]. The aim is to obtain the largest increase in the objective function so that more nodes can be pruned later on.

Another branching rule which does not require any problem specific knowledge but is based on pseudocosts has been introduced by Benichou et al. [2]. They compute so-called pseudocosts which estimate the change in the objective induced by the branching. Gupta and Ravindran [13] report numerical experience with pseudocosts

for a range of MINLP test problems and conclude that it is inferior to the branching rule which is described in the previous paragraph.

Körner [15] proposes a branching rule which aims to minimize the size of the tree so that fewer QP problems need be solved. He shows how the tree size resulting from a particular branching order can be estimated efficiently. The drawback of these estimates is that they are based on a special tree structure. Instead of branching (on x_i) by creating two problems, a range of problems is created with fixed values $x_i = i \forall i = l_i, l_i+1, \dots, u_i$. This branching structure can lead to wide trees, especially if the simple bounds are not very strong.

3. Improved lower bounds for MIQP branch-and-bound. A lower bound for both child nodes is readily available as the optimal value of the parent problem, since $\mathcal{P}$ is convex. However, these bounds are usually very weak and rarely give rise to any fathoming at a later stage once an integer feasible solution has been obtained. Improved lower bounds have therefore been considered in [1] for MILP and are readily derived in the context of an active set method (ASM) (e.g., [6]) for MILP problems.

This section considers MIQP problems and shows how improved lower bounds can be derived for the child nodes of $\mathcal{P}'$, where $\mathcal{P}'$ is the QP relaxation of $\mathcal{P}$ as defined in section 2. The procedure can be readily applied to any other node of the MIQP branch-and-bound tree. Let the solution to $\mathcal{P}'$ be x' , and assume that x'_j , $j \in I$ is fractional. Define $\mathcal{P}^-$ as the QP obtained from $\mathcal{P}'$ by branching down (i.e., by adding the simple bound $x_j \leq \lfloor x'_j \rfloor$) and $\mathcal{P}^+$ as the QP obtained from $\mathcal{P}'$ by branching up (i.e., by adding $x_j \geq \lceil x'_j \rceil$).

The derivation of lower bounds is best explained using a tableau representation of the dual ASM (e.g. [7]). This tableau is *implicitly* available after the parent QP has been solved with any ASM-like QP solver. In an active set method the constraints are divided into two disjoint sets, the *active set* $\mathcal{A}$ and the set $\mathcal{I}$ of inactive constraints. If no degeneracy occurs, then the active set $\mathcal{A}$ can be identified with the set of active constraints. Partitioning the multipliers λ and the residual r of the constraints (including the simple bounds) accordingly, the tableau then defines basic or dependent variables $\lambda_{\mathcal{A}}$ and $r_{\mathcal{I}}$ in terms of independent or nonbasic variables $r_{\mathcal{A}}$ and $\lambda_{\mathcal{I}}$.

In general then, the corresponding linear complementarity problem (LCP) tableau (e.g., [7]) has the form

$$(3.1) \quad \begin{array}{cc} & \begin{array}{cc} r_{\mathcal{A}} & \lambda_{\mathcal{I}} \end{array} \\ \begin{array}{c} \lambda_{\mathcal{A}} \\ r_{\mathcal{I}} \end{array} & \left[\begin{array}{cc} G' & A' \\ -A'^T & H' \end{array} \right] \left[\begin{array}{c} \lambda'_{\mathcal{A}} \\ r'_{\mathcal{I}} \end{array} \right], \end{array}$$

where the current value of the basic variables is given by the right-hand side $(\lambda'_{\mathcal{A}}, r'_{\mathcal{I}})$ and the current value of the nonbasic variables is zero. Columns in the tableau indicate how the basic variables are affected by changes in the nonbasic variables. It is possible to derive expressions for A' , G' , and H' and these are given towards the end of this section in the case that the QP solver employs a null-space method.

First it is described how the dual ASM can be used to derive improved lower bounds for the case that less than n constraints are active at the solution of the parent problem $\mathcal{P}'$. Any dual feasible point for $\mathcal{P}^-$ or $\mathcal{P}^+$ would provide a lower bound on the optimum of $\mathcal{P}^-$ or $\mathcal{P}^+$. The solution of the parent problem $\mathcal{P}'$ is dual feasible in both $\mathcal{P}^-$ and $\mathcal{P}^+$. However, the lower bound derived from this point rarely gives rise to much fathoming, and we therefore wish to improve this lower bound.

In order to improve the lower bound on $\mathcal{P}^-$ and $\mathcal{P}^+$ derived from the optimal solution of the parent problem $\mathcal{P}'$, we consider one step of the dual ASM. Problems

$\mathcal{P}^-$ and $\mathcal{P}^+$ differ from $\mathcal{P}'$ by the addition of a single simple bound which makes the primal solution of $\mathcal{P}'$ infeasible in $\mathcal{P}^-$ and $\mathcal{P}^+$. The method that is described here can be seen as moving parametrically from x' towards $[x'_j]$ (or $[x'_j + 1]$) while maintaining dual feasibility. The move is completed when a pivot would be required in the dual ASM as we wish to avoid the expense of this computation.

The dual ASM of Goldfarb and Idnani [12] starts with a dual feasible point. This makes the method very amenable to MIQP branch-and-bound since upon solving the parent problem its solution (x', λ') is dual feasible in both child problems. The dual ASM takes one constraint that is violated and tries to increase its multiplier. As a consequence of the tableau structure the increase of the multiplier reduces the primal infeasibility while maintaining all other feasibilities.

All necessary quantities for the step can be computed from the final LCP tableau of the parent problem. This tableau is either available directly or indirectly (as in our case) if the parent problem is solved with an ASM. Assume that the final LCP tableau of the parent problem is given by (3.1). Since x'_j is fractional, it follows that $j \in \mathcal{I}$, or in other words x'_j is not at a simple bound. Let λ_j be the complementary variable of the simple bound constraint on x_j . The dual ASM aims to increase λ_j away from 0 until one of the other multipliers becomes 0. This corresponds to moving the bound of x_j parametrically towards its new value, $[x'_j]$ say. The aim is to determine how far the bound can be moved in the null space of the active constraints without destroying dual feasibility.

The effect of increasing λ_j on the multiplier of the active constraints is determined by the column of the LCP tableau (3.1) corresponding to λ_j . In particular the effect on the multipliers of the active constraints is

$$\lambda_{\mathcal{A}} = \lambda'_{\mathcal{A}} - \alpha a'_j$$

for moving x_j towards $[x'_j + 1]$ and

$$\lambda_{\mathcal{A}} = \lambda'_{\mathcal{A}} + \alpha a'_j$$

for moving x_j towards $[x'_j]$, where a'_j is the column of A' corresponding to λ_j . Also λ_j itself is increased from 0 to α . The maximum step α that can be taken without losing dual feasibility is determined by the dual ratio test. Thus

$$\alpha^+ = \min_{i : a'_{ij} > 0} \left(\frac{\lambda'_i}{a'_{ij}} \right)$$

gives the dual feasible steplength for moving towards $[x'_j + 1]$ and

$$\alpha^- = \min_{i : a'_{ij} < 0} \left(-\frac{\lambda'_i}{a'_{ij}} \right)$$

gives the dual feasible steplength in the second case. If there exists no $a'_{ij} > 0$ or no $a'_{ij} < 0$, respectively, then α is set to ∞ , since the dual is unbounded. In this case it follows that the primal is infeasible and the respective dual node is therefore fathomed. This particular child problem is therefore not added to the list of pending problems. Similarly, if the improved lower bound is greater than or equal to the current upper bound, the child problem is not added to the list of pending problems.

Finally the lower bounds on $\mathcal{P}^+$ and $\mathcal{P}^-$ can be computed. Let s be the search direction of the dual ASM; i.e., $s_i = [h'_j]_i$ if $i \in \mathcal{I}$, $i \leq n$ and $s_i = 0$ otherwise, where

h'_j is column j of H' . If $\phi_j = x_j - [x_j]$ is the fractional part of x_j and if f' is the optimal value of the parent, then the lower bounds L^+ and L^- , respectively, are given by the values of the Lagrangian at the new points

$$L^+ = f' + \frac{1}{2}(\alpha^+)^2 s^T G s + \alpha^+(1 - \phi_j - \alpha^+ h'_{jj}),$$

$$L^- = f' + \frac{1}{2}(\alpha^-)^2 s^T G s + \alpha^-(\phi_j - \alpha^- h'_{jj}).$$

There are two terms contributing to the improved bounds. First there is the quadratic term corresponding to the second order change in the objective. The lower bound is further improved by the last term which corresponds to a Lagrangian-type penalty for the violation of the new simple bound, namely the product of the multiplier α of the new simple bound with the amount by which the simple bound is not yet satisfied.

In the special case that n constraints are active at the solution of $\mathcal{P}'$, then the final LCP tableau can be simplified to

$$\begin{array}{cc} \lambda_{\mathcal{A}} & \begin{array}{cc} r_{\mathcal{A}} & \lambda_{\mathcal{I}} \\ G' & A' \end{array} \\ r_{\mathcal{I}} & \begin{bmatrix} -A'^T & 0 \end{bmatrix} \end{array} \quad \begin{bmatrix} \lambda'_{\mathcal{A}} \\ r'_{\mathcal{I}} \end{bmatrix},$$

where A' is an (nonsingular) $n \times n$ matrix. In this case $h'_j = 0$ and the lower bounds are $L^+ = f' + \alpha^+(1 - \phi_j)$ for $\mathcal{P}^+$ and $L^- = f' + \alpha^-\phi_j$ for $\mathcal{P}^-$. In the case where n constraints are active at the solution of the QP it is possible to derive the same lower bounds using marginal costs p_j^+ and p_j^- and the convexity of f similar to [6, page 337].

In the remainder of this section we indicate how the lower bound procedure can be implemented and give an operation count. The QP solver employed in the present implementation uses a primal active set method where equality QP problems are solved by a null-space method [7].

At the end of each solve a partition of the constraint matrix A (which now includes also the simple bounds) into active and inactive constraints is available. However, similar information is likely to be available with other solvers, for instance with solvers based on linear complementarity problems. It is then possible to derive expressions for G' , A' , and H' ([7, Theorem 2.1]).

- $A' = T^T A_{\mathcal{I}}$, and the off-diagonal blocks in (3.1) are skew symmetric.
- $G' = U$ is symmetric and negative definite if G is positive definite.
- $H' = -A_{\mathcal{I}}^T H A_{\mathcal{I}}$ is symmetric and negative semidefinite.

The matrices H , T , and U are defined through the inverse

$$(3.2) \quad \begin{bmatrix} G & -A_{\mathcal{A}} \\ -A_{\mathcal{A}}^T & 0 \end{bmatrix}^{-1} = \begin{bmatrix} H & -T \\ -T^T & U \end{bmatrix}.$$

At the end of the QP solve of the parent problem we assume that LU factors of the basis matrix $[A_{\mathcal{A}} : V]$ are available, where $A_{\mathcal{A}}$ is the matrix whose columns are the active constraint normals and V is any matrix such that $[A_{\mathcal{A}} : V]$ is nonsingular. This includes QR factors of $A_{\mathcal{A}}$ but also the important practical case where V is made up of columns of the unit matrix. If

$$[A_{\mathcal{A}} : V]^{-1} = \begin{bmatrix} Y^T \\ Z^T \end{bmatrix},$$

then it is possible to obtain expressions for H , T , and U (e.g., [6]). These lead to explicit expressions for A' and H' , given by

$$\begin{aligned} H' &= -A_{\mathcal{I}}^T Z (Z^T G Z)^{-1} Z^T A_{\mathcal{I}}, \\ A' &= (Y - Z (Z^T G Z)^{-1} Z^T G Y)^T A_{\mathcal{I}}, \end{aligned}$$

where the “reduced Hessian” $(Z^T G Z)$ is available *implicitly* through LDL^T factors at the end of the solve of the parent QP. Note that it is not necessary to form $(Z^T G Z)$ explicitly. Instead we update factors of the reduced Hessian matrix as the basis changes.

Thus it is possible to derive an expression for a'_j .

$$a'_j = A' e_j = Y^T \underbrace{A_{\mathcal{I}} e_j}_{e_j} - Y^T G Z (Z^T G Z)^{-1} Z^T \underbrace{A_{\mathcal{I}} e_j}_{e_j}.$$

$\underbrace{\hspace{10em}}_v$
 $\underbrace{\hspace{10em}}_w$
 $\underbrace{\hspace{10em}}_u$

A similar expression can be derived for $h'_j = H' e_j$. Note that the intermediate vector u is used both for h'_j and a'_j .

In the case where fewer than n constraints are active at the solution of $\mathcal{P}'$ the following computations are carried out to derive the lower bound. We also give the flop count in the case that dense matrix factorizations are employed (here $k = \dim(\text{null}(A_{\mathcal{A}}))$).

1. Form $v = Z^T e_j$ by performing 1 solve with the LU factors of $[A_{\mathcal{A}} : V]^{-1}$. (n^2 flops)
2. Use LDL^T factors of the reduced Hessian to form $w = (Z^T G Z)^{-1} v$. (k^2 flops)
3. Form $u = Zw$ using the LU factors of $[A_{\mathcal{A}} : V]$. (n^2 flops)
4. Compute m scalar products to form $h'_j = -A_{\mathcal{I}}^T u$. (mn flops)
5. Form the matrix vector product $u = e_j - Gu$. (n^2 flops)
6. Form $a'_j = Y^T u$ using the LU factors of $[A_{\mathcal{A}} : V]$. (n^2 flops)
7. Carry out the ratio test to find α^+ and α^- . ($n + 2$ flops)
8. Compute the lower bounds L^+ and L^- . ($n^2 + n + 4$ flops)

Summing the individual operation counts shows that the lower bounds can be computed in $5n^2 + mn + k^2 + \mathcal{O}(n)$ flops. Note that the operation count for step 4 is overly pessimistic. Many inactive constraints will usually be simple bounds for which the scalar product in 4 costs only 1 and not n flops. It is worth mentioning that sparse matrix techniques for factoring and for representing $[A_{\mathcal{A}} : V]$ can significantly reduce the n^2 component in the operation count. In particular, if V is made up of unit columns this operation count can be reduced to $\mathcal{O}((n - k)^2)$ flops. We do not include the cost for forming LU factors in the flop count since these factors are available if the parent QP has been solved by an ASM.

We are grateful to an anonymous referee for pointing out that if range space factors were used to solve the parent problem, then the computation of a'_j in step 6 would be cheaper. However, range space factors are less convenient for handling upper and lower bounds on the variables in a QP solver and the factors of the Hessian matrix G may be expensive to compute or not even exist.

The procedure to compute a lower bound is considerably cheaper than a QP solve, but it is an order of magnitude more expensive than the simpler branching

rules based upon the most fractional variable. Only extensive numerical testing can reveal whether or not this additional effort results in an overall reduction in CPU time for MIQP problems and this together with a small illustrative example is considered in the next section.

4. A small example and numerical experience. It is instructive to consider applying the above procedure to a simple example:

$$\mathcal{P} \begin{cases} \min_x & x_1^2 - x_1x_2 + x_2^2 - 3x_1 \\ \text{subject to} & -x_1 - x_2 \geq -2, \\ & x \geq 0, \ x_1 \text{ integer.} \end{cases}$$

The solution of $\mathcal{P}$ with its integer restrictions relaxed is $x' = (\frac{3}{2}, \frac{1}{2})$, $f' = -\frac{99}{36}$ and the final LCP tableau is given by

$$\begin{array}{c} r_1 \quad \pi_1 \quad \pi_2 \\ \lambda_1 \quad \left[\begin{array}{c|cc} -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} \\ \hline \frac{1}{2} & -\frac{1}{6} & \frac{1}{6} \\ \frac{1}{2} & \frac{1}{6} & -\frac{1}{6} \end{array} \right] \quad \left[\begin{array}{c} \frac{1}{2} \\ \frac{3}{2} \\ \frac{1}{2} \end{array} \right], \\ x_1 \\ x_2 \end{array}$$

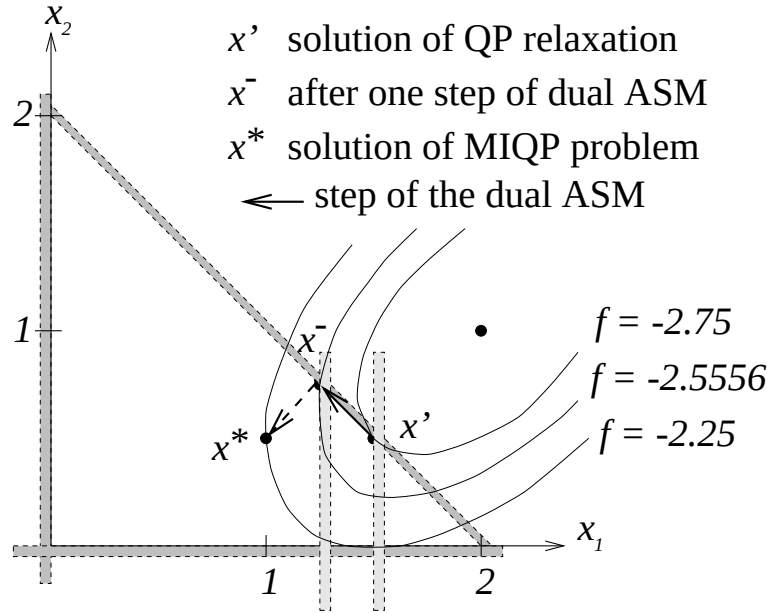
where π_i denotes the complementary variable to x_i . Let $\mathcal{P}^-$ denote the problem obtained from $\mathcal{P}$ by adding the new upper bound $x_1 \leq 1$. To compute a lower bound on the solution of $\mathcal{P}^-$ the dual ratio test is carried out, giving $\alpha^- = \min(-\frac{1}{2}/-\frac{1}{2}) = 1$. The search direction in terms of the x variables is $s = (-\frac{1}{6}, \frac{1}{6})^T$ and the step of the dual active set method moves to the point $x^- = x' + \alpha^- h' = (\frac{4}{3}, \frac{2}{3})^T$ which results in the improved lower bound of $L^- = -\frac{84}{36}$. This improvement corresponds to closing 80 percent of the gap between the solution values of $\mathcal{P}$ and $\mathcal{P}^-$ (which is $-\frac{81}{36}$). Similarly the lower bound for $\mathcal{P}^+$ can be computed as $L^+ = -2$. The step of the dual ASM is illustrated in Figure 4.1.

It is indicated in section 3 that improved lower bounds are not cheaply computed and it is therefore important to carry out numerical testing to assess to what extent the savings in the number of problems solved are mirrored by savings in the CPU time. To this end three different branching rules have been implemented which make use of the improved lower bounds in different ways. BR1 branches on the most fractional integer variable first and uses no improved lower bounds at all. BR2 computes lower bounds L_j^+ and L_j^- for all fractional integer variables and branches on the integer variable that satisfies

$$\max_j \min(L_j^+, L_j^-).$$

Finally, BR3 is a hybrid of BR1 and BR2, designed to expose the effect of the new bounds. It uses branching rule BR1 to determine the branching variable, but in addition computes lower bounds for the two child problems. Each branching rule requires a different number of floating point operations every time it is applied. BR1 requires $\mathcal{O}(n)$, BR3 requires $\mathcal{O}(n^2)$, and BR2 requires $\mathcal{O}(n^3)$ flops per application. An alternative to BR3 is to solve the child problems by the dual ASM and use a cut-off value to terminate the solve (BR3 is in fact equivalent to one dual ASM iteration).

The test problems come from a variety of backgrounds. “Beale” and “HS76” are problems found in the literature (e.g., [14]). The problems “AVGAS,” “AFIRO,” and

FIG. 4.1. *Illustration of the dual active set method.*

“SHARE2B” are originally LP problems which can be found in, e.g., [24] and the SOL test set, respectively. These LP problems are modified by adding integer restrictions and by replacing the linear objective function by a quadratic objective with positive definite Hessian matrix. QAP(6,4) is a quadratic assignment problem for allocating four plants to six locations. Finally, “portfolio 1” and “portfolio 2” are problems that arise in the index tracking for passive portfolio management. In both cases a portfolio of one and two investments out of 17 possible investments is selected.

Table 4.1 gives a brief description of the test problems. The first column gives the name of the problem followed by four columns that detail the dimensions of the problems; n_i is the number of integer variables, n_c is the number of continuous variables, m_e gives the number of equality constraints, and finally m_i details the number of inequality constraints. f' is the optimal value of the QP relaxation and f^* is the optimal value of $\mathcal{P}$. The last column gives the range of the integer variables x_I .

The following tables give the results of the test run on a SUN-SPARC SLC using the FORTRAN 77 compiler with the `-fast` option. All problems are solved in single precision FORTRAN to a “mixed accuracy” of $\epsilon = 10^{-6}$; that is, an approximate solution $\hat{x}$ is found for which $|f(\hat{x}) - f^*| \leq (1 + |f^*|) \cdot \epsilon$. The QP problem at each node is solved using `bqp`, which implements a primal ASM and resolves degeneracy giving a guaranteed termination even in the presence of round-off errors [7]. Table 4.2 gives the number of QP problems generated by each of the branching rules, the number of QP problems solved, and the percentage of problems solved. The CPU time needed by each branching rule and for the three other solvers is presented in Table 5.2 in the next section. Runs which took more than 2,000 seconds CPU time were abandoned.

BR3 and BR1 use the same branching rule. Therefore, comparing their respective results in Table 4.2 shows for BR1 that very little pruning takes place if the lower bounds are taken as the optimal value of the parent problem. BR3 in comparison solves a lower percentage of the problems that are generated. The difference in the

TABLE 4.1
Description of MIQP test problems.

Problem	n_i	n_c	m_e	m_i	f'	f^*	Integer Range
Beale	3	0	0	1	-8.8889	-8.0	$0 \leq x_I \leq 5$
HS76	4	0	0	3	-4.6818	-4.5	$0 \leq x_I \leq 5$
AVGAS (1a)	8	0	0	10	-4.6319	-1.5	$0 \leq x_I \leq 1$
AVGAS (1b)	8	0	0	10	-4.4832	-1.0	$0 \leq x_I \leq 1$
AFIRO (1a)	26	6	8	19	1048.01	1205.5	$0 \leq x_I \leq 5$
AFIRO (1b)	26	6	8	19	1081.14	1205.5	$0 \leq x_I \leq 5$
AFIRO (2a)	19	13	8	19	1048.01	1068.2	$0 \leq x_I \leq 5$
AFIRO (2b)	19	13	8	19	1074.52	1099.5	$0 \leq x_I \leq 5$
AFIRO (2c)	19	13	8	19	64.21	66.96	$0 \leq x_I \leq 5$
SHARE2B (8)	8	71	12	84	2977.67	3132.6	$0 \leq x_I \leq 5$
SHARE2B (12)	12	71	12	84	2977.67	3159.7	$0 \leq x_I \leq 5$
SHARE2B (24)	24	59	12	84	2977.67	3332.8	$0 \leq x_I \leq 5$
QAP(6,4)	24	55	4	6	$6.21 \cdot 10^{-3}$	3.0	$0 \leq x_I \leq 1$
portfolio 1	17	17	2	17	$-8.59 \cdot 10^{-4}$	$-7.11 \cdot 10^{-4}$	$0 \leq x_I \leq 1$
portfolio 2	17	17	2	17	$-8.59 \cdot 10^{-4}$	$-7.92 \cdot 10^{-4}$	$0 \leq x_I \leq 1$

TABLE 4.2
Number of continuous problems generated and solved by branch-and-bound.

Problem	BR1			BR2			BR3		
	generated	solved	% solved	generated	solved	% solved	generated	solved	% solved
Beale	7	7	100	7	4	57	7	4	57
HS76	5	5	100	5	3	60	5	3	60
AVGAS (1a)	21	20	95	19	14	74	20	13	65
AVGAS (1b)	19	19	100	29	17	59	19	13	68
AFIRO (2a)	9	9	100	11	7	64	8	5	63
AFIRO (2b)	17	17	100	17	13	76	14	11	79
AFIRO (2c)	39	35	90	31	19	61	45	30	67
AFIRO (1a)	3,439	2,948	86	17	15	88	3,180	2,225	70
AFIRO (1b)	1,125	900	80	29	27	93	985	684	69
SHARE2B (8)	65	57	88	15	9	60	65	45	69
SHARE2B (12)	729	708	97	49	29	59	645	396	61
SHARE2B (24)	> 2000 s CPU			304	185	61	> 2000 s CPU		
QAP(6,4)	319	319	100	397	353	89	319	289	91
portfolio 1	33	26	79	31	26	84	33	25	76
portfolio 2	171	168	98	209	180	86	173	136	79

number of problems generated for the larger examples can easily be explained. The effect of the improved pruning for BR3 is that both branching rules search the tree in a slightly different order. As one would expect, the lower bounds result only in a modest improvement in terms of the number of QP problems solved (this would be similar if the QP problems were solved with a dual ASM and a cut-off value). Note that one cannot expect to prune more than 50 percent of the nodes through lower bounds, which limits the scope for improvement through pruning for the hybrid branching rule BR3.

An interesting feature is that there exist difficult MIQP problems (such as AFIRO and SHARE2B) where BR2 improves the performance of the branch-and-bound scheme dramatically. With BR2 only a fraction of the number of QPs that are solved using BR1 or BR3 are solved. This reduction in the number of QPs translates into a reduc-

tion in terms of CPU time resulting in an order of magnitude reduction in the CPU time of BR2 compared to the other two branching rules for these problems.

This improvement appears to be due to the ability of BR2 to find an integer feasible solution faster than the two other methods. For example, for AFIRO (1a) BR2 finds the optimal solution after only 15 QP solves whereas BR1 and BR3 require 2,348 and 2,143 QP solves, respectively, before the first integer feasible solution is found. Effectively, BR2 explores the neighborhood of the solution of the parent problem in the direction in which branching is possible, and this seems to improve the decision process that selects the next branching variable in a dramatic manner.

However, this observation appears to be true only for MIQP problems for which the notion of “neighborhood” has a sensible meaning such as general MIQP problems. In particular, this is not true for 0–1 QPs and this explains the rather disappointing performance of BR2 on “AVGAS” and especially “QAP(6,4)” and “portfolio”. For the latter two problems BR2 solves about 10 percent more QP problems which results in a doubling of the CPU times compared to BR1 and BR3 (which are the cheaper branching rules).

5. Numerical experience with other solvers. In this section we report on experience which we obtained with other MIQP solvers. The solvers we implemented are based on generalized Benders decomposition (GBD) (e.g., [11], [18], and [9]), outer approximation (OA) (e.g., [5], [8]), and LP/QP based branch-and-bound [21]. All these solvers have in common that they solve the MIQP problem via a finite sequence of MILP master program relaxations and QP subproblems obtained by fixing the integer variables. The master program is obtained by projection onto the integer variables, together with dualization (for GBD) and outer approximation for OA. The LP/QP based branch-and-bound algorithm differs in the way in which the related master programs are solved. Instead of solving a sequence of successive MILP master program relaxations as for OA, the branch-and-bound process for solving the MILP master program is modified. Whenever an integer assignment is generated during the tree search, the corresponding QP subproblem is solved and new OAs are added to all pending LP problems. Thus instead of resolving the master program, it is effectively updated. All three solvers were again implemented in single precision FORTRAN 77 using `bqp` to solve the QP subproblems and an MILP version of the MIQP branch-and-bound solver of the previous section to solve the master problems. In order to gain a fair comparison the best branching rule for each problem was used in the tests.

Table 5.1 shows the performance of GBD, OA, and LP/QP based branch-and-bound on the test problems. The table shows the number of iterations (or QP solves) and the number of LP solves for each problem.

A comparison of the three other solvers with branch-and-bound shows that branch-and-bound is the clear winner both in terms of the number of problems solved and in terms of CPU time. With few exceptions branch-and-bound is an order of magnitude faster than any of the other solvers (and never slower). Moreover, the three relaxation based solvers failed to solve a number of problems within the time limit of 2,000 seconds CPU time. The reason for the poor performance of GBD, OA, and LP/QP based branch-and-bound is that while they solve fewer QP problems the number of LP problems grows dramatically.

Finally, LP/QP based branch-and-bound is seen to be more efficient than OA or GBD since it avoids the resolution of successive MILP master programs and solves only a few more QP problems. GBD performs best on problems which have relatively few integer variables since its master problem involves only the integer variables (plus

TABLE 5.1
Number of iterations and number of LPs solved by OA, GBD, and LP/QP.

Problem	OA		GBD		LP/QP	
	iter	LPs	iter	LPs	iter	LPs
Beale	7	47	7	41	6	20
HS76	2	14	6	38	2	12
AVGAS (a)	10	195	13	145	10	42
AVGAS (b)	11	190	15	151	11	47
AFIRO (2a)	12	846	22	975	23	236
AFIRO (2b)	14	858	24	748	18	145
AFIRO (2c)	69	9,826	64	10,054	31	203
AFIRO (1a)	4	2,353	16	1,566	4	429
AFIRO (1b)	5	548	24	10,865	4	26
SHARE2B (8)	10	463	34	1,202	13	255
SHARE2B (12)	> 2000 s CPU		97	21,217	> 2000 s CPU	
SHARE2B (24)	> 2000 s CPU		> 2000 s CPU		> 2000 s CPU	
QAP(6,4)	361	9,116	373	8,976	361	935
portfolio 1	18	299	20	227	18	50
portfolio 2	94	8,159	122	5,952	98	356

TABLE 5.2
CPU times (seconds) for branch-and-bound, OA, GBD, and LP/QP.

Problem	BR1	BR2	BR3	OA	GBD	LP/QP
Beale	0.00	0.01	0.02	0.16	0.14	0.07
HS76	0.01	0.02	0.01	0.06	0.15	0.06
AVGAS (1a)	0.16	0.16	0.14	2.42	1.26	0.59
AVGAS (1b)	0.14	0.17	0.14	2.26	1.33	0.63
AFIRO (2a)	0.69	0.71	0.49	113.19	49.42	69.45
AFIRO (2b)	0.90	1.02	0.70	118.98	36.92	52.55
AFIRO (2c)	2.64	2.06	2.36	836.30	362.64	30.02
AFIRO (1a)	112.49	1.70	92.79	126.73	49.21	35.64
AFIRO (1b)	32.60	3.19	28.56	31.69	440.77	4.09
SHARE2B (8)	37.54	13.09	31.76	470.93	125.33	907.43
SHARE2B (12)	418.70	29.96	238.87	> 2000	1420.74	> 2000
SHARE2B (24)	>2000	207.29	>2000	> 2000	> 2000	> 2000
QAP(6,4)	9.87	15.54	9.63	1159.01	1529.23	139.96
portfolio 1	1.26	2.76	1.29	24.77	7.21	8.32
portfolio 2	9.58	18.27	8.15	> 2000	234.82	69.87

one dummy continuous variable). However, none of the three alternatives to branch-and-bound seem able to compete with branch-and-bound which is often an order of magnitude faster than any of the alternatives.

It should not come as a surprise that none of the solvers presented in this section are comparable to branch-and-bound. The reason for this is that the solution of a QP child problem is very cheap (the average number of pivots required to solve a child problem is between one and two for “AFIRO” and between four and 10 for “SHARE2B”) resulting in an inexpensive tree search (comparable to the MILP situation). However, this situation does not hold for MINLP branch-and-bound where each node may require the solution of several QP problems. Our own experience [19] shows that there exist important classes of MINLP problems for which branch-and-bound is not the best method.

6. Conclusions. It has been shown how improved lower bounds for MIQP branch-and-bound can be computed by parametrically taking one step of the dual active set method. A modest improvement through the use of improved lower bounds

for a standard branching rule is observed. An interesting point is that on some hard problems a branching rule based entirely on the new bounds achieved an improvement of an order of magnitude compared to a standard branching rule. The branch-and-bound solver has also been compared to three alternative solvers based on relaxing an equivalent MILP master program. The numerical tests show that these solvers are an order of magnitude slower than branch-and-bound.

REFERENCES

- [1] E. M. L. BEALE, *Integer programming*, in The State of the Art in Numerical Analysis, D. A. H. Jacobs, ed., Academic Press, London, 1978.
- [2] M. BENICHO, J. M. GAUTHIER, P. GIRODET, G. HENTGES, G. RIBIERE, AND O. VINCENT, *Experiments in mixed-integer linear programming*, Math. Programming, 1 (1971), pp. 76–94.
- [3] R. BREU AND C.-A. BURDET, *Branch-and-bound experiments in 0-1 programming*, Math. Programming Study, 2 (1974), pp. 1–50.
- [4] R. J. DAKIN, *A tree search algorithm for mixed integer programming problems*, Comput. J., 8 (1965), pp. 250–255.
- [5] M. DURAN AND I. E. GROSSMANN, *An outer-approximation algorithm for a class of mixed-integer nonlinear programs*, Math. Programming, 36 (1986), pp. 307–339.
- [6] R. FLETCHER, *Practical Methods of Optimization*, 2nd ed., John Wiley, Chichester, 1987.
- [7] R. FLETCHER, *Resolving degeneracy in quadratic programming*, Ann. Oper. Res., 47 (1993), pp. 307–334.
- [8] R. FLETCHER AND S. LEYFFER, *Solving mixed integer nonlinear programs by outer approximation*, Math. Programming, 66 (1994), pp. 327–349.
- [9] O. E. FLIPPO AND A. H. G. RINNOY KAN, *A note on Benders decomposition in mixed-integer quadratic programming*, Oper. Res. Lett., 9 (1990), pp. 81–83.
- [10] R. S. GARFINKEL AND G. L. NEMHAUSER, *Integer Programming*, John Wiley, New York, 1972.
- [11] A. M. GEOFFRION, *Generalized Benders decomposition*, J. Optim. Theory Appl., 10 (1972), pp. 237–260.
- [12] D. GOLDFARB AND A. IDNANI, *A numerically stable dual method for solving strictly convex quadratic programs*, Math. Programming, 27 (1983), pp. 1–33.
- [13] O. K. GUPTA AND A. RAVINDRAN, *Branch and bound experiments in convex nonlinear integer programming*, Management Sci., 31 (1985), pp. 1533–1546.
- [14] W. HOCK AND K. SCHITTKOWSKI, *Test Examples for Nonlinear Programming Codes*, Springer-Verlag, Berlin, 1981.
- [15] F. KÖRNER, *A new branching rule for the branch-and-bound algorithm for solving nonlinear integer programming problems*, BIT, 28 (1988), pp. 701–708.
- [16] J. W. KWIATOWSKI, *Algorithms for Index Tracking*, Working paper 90.1, University of Edinburgh, Centre for Financial Markets Research, Edinburgh, UK, 1990.
- [17] A. H. LAND AND A. G. DOIG, *An automatic method for solving discrete programming problems*, Econometrica, 28 (1960), pp. 497–520.
- [18] R. LAZIMY, *Improved algorithm for mixed-integer quadratic programs and a computational study*, Math. Programming, 32 (1985), pp. 100–113.
- [19] S. LEYFFER, *Deterministic Methods for Mixed Integer Nonlinear Programming*, Ph.D. thesis, University of Dundee, Dundee, Scotland, UK, 1993.
- [20] M. PADBERG AND G. RINALDI, *A branch-and-cut algorithm for the resolution of large-scale symmetric travelling salesman problems*, SIAM Rev., 33 (1991), pp. 60–100.
- [21] I. QUESADA AND I. E. GROSSMANN, *An LP/NLP based branch-and-bound algorithm for convex MINLP optimization problems*, Comput. Chemical Engineering, 16 (1992), pp. 937–947.
- [22] H. A. TAHA, *Integer Programming – Theory, Applications and Computations*, Academic Press, London, 1975.
- [23] O. V. VOLKOVICH, V. R. ROSHCHIN, AND I. V. SERGIENKO, *Models and methods of solution of quadratic integer programming problems*, Cybernetics, 23 (1987), pp. 289–305.
- [24] A. C. WILLIAMS, *Computational experience with ellipsoid algorithms for linear programming*, Math. Programming Soc., COAL Newsletter, 5 (1981), pp. 37–48.

BICRITERION SINGLE MACHINE SCHEDULING WITH RESOURCE DEPENDENT PROCESSING TIMES*

T. C. EDWIN CHENG[†], ADAM JANI[‡], AND MIKHAIL Y. KOVALYOV[§]

Abstract. A bicriterion problem of scheduling jobs on a single machine is studied. The processing time of each job is a linear decreasing function of the amount of a common discrete resource allocated to the job. A solution is specified by a sequence of the jobs and a resource allocation. The quality of a solution is measured by two criteria, F_1 and F_2 . The first criterion is the maximal or total (weighted) resource consumption, and the second criterion is a regular scheduling criterion depending on the job completion times. Both criteria have to be minimized. General schemes for the construction of the Pareto set and the Pareto set ϵ -approximation are presented. Computational complexities of problems to minimize F_1 subject to $F_2 \leq K$ and to minimize F_2 subject to $F_1 \leq K$, where K is any number, are studied for various functions F_1 and F_2 . Algorithms for solving these problems and for the construction of the Pareto set and the Pareto set ϵ -approximation for the corresponding bicriterion problems are presented.

Key words. single machine scheduling, resource allocation, bicriterion scheduling, approximation

AMS subject classifications. 68Q25, 90C39

PII. S1052623495288192

1. Introduction. Scheduling problems with resource dependent job parameters could be found in many practical settings (see, for example, Williams (1985) and Janiak (1991)). The study of single machine scheduling problems with resource dependent job processing times was initiated by Vickson (1980a), (1980b). A survey of the results for this class of problems was provided by Nowicki and Zdrzalka (1990). Most research in this area has focussed on single criterion problems. In practice, however, quality is a multidimensional concept (Willborn and Cheng (1994)) and so it is apposite to study scheduling problems with multicriterion objective functions. Bicriterion single machine scheduling problems with fixed job parameters have been well studied in recent years. Some remarkable work in this area has been done by Hoogeveen (1992) and Lee and Vairaktarakis (1993).

There are many practical scheduling situations in which the effectiveness of a schedule can be improved through an adequate allocation of resources to the jobs to be processed by a facility. For example, in project scheduling, the project-completion time can be compressed if additional resources are allocated to speed up the processing of the critical tasks. However, in reality, resources are usually costly production factors limited in supply. Therefore, a firm aims either to minimize the resource consumption subject to a given level of service or to maximize the service level subject to some resource constraints. In this paper, we study the bicriterion single machine scheduling problem with linear resource dependent job processing times.

*Received by the editors June 26, 1995; accepted for publication (in revised form) January 8, 1997.

<http://www.siam.org/journals/siopt/8-2/28819.html>

[†]Office of the Vice President (Research and Postgraduate Studies), The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong (mscheng@polyu.edu.hk). The research of this author was supported in part by university research grant 0351-193-A3-014.

[‡]Institute of Engineering Cybernetics, Technical University of Wrocław, 50370 Wrocław, Poland (janiak@sequoia.ict.pwr.wroc.pl).

[§]Institute of Engineering Cybernetics, Belarus Academy of Sciences, 220012 Minsk, Belarus. The research of this author was supported in part by INTAS grant 93-257.

The problem may be stated as follows. There are n independent nonpreemptive jobs to be processed on a single machine and a single discrete resource which can be allocated to jobs. Each job j becomes available for processing at time zero and has a due date d_j and a resource dependent processing time

$$p_j = b_j - a_j x_j.$$

Here b_j is the normal processing time of job j that can be compressed by an amount of $a_j x_j$ if x_j units of a resource are allocated to this job; a_j is the unit processing time compression for job j . There is a limit on the amount x_j of the resource that can be allocated to job j :

$$x_j \in \{0, 1, \dots, \tau_j\}.$$

Due to the nonnegativity of processing times, $\tau_j \leq b_j/a_j$ is assumed for $j = 1, \dots, n$.

A solution is specified by a sequence of the jobs and a resource allocation $(x_1, \dots, x_n)$. For any solution, the completion time C_j of each job j is easily determined. The quality of a solution is measured by two criteria, F_1 and F_2 . The first criterion F_1 is the maximal or total (weighted) resource consumption, and the second criterion F_2 is a regular scheduling criterion depending on the job completion times. Both criteria have to be minimized.

Two weights v_j and w_j are associated with each job j . A weight v_j indicates a relative importance of job j with respect to a resource consumption criterion, while a weight w_j indicates its relative importance with respect to some scheduling criterion. We consider $F_1 \in \{g_{\max}, \sum x_j, \sum v_j x_j\}$ and $F_2 \in \{f_{\max}, C_{\max}, \sum U_j, \sum w_j U_j, \sum C_j, \sum w_j C_j\}$, where $U_j = 0$ if $C_j \leq d_j$ and $U_j = 1$ otherwise, $g_{\max} = \max\{g_j(x_j)\}$ and $f_{\max} = \max\{f_j(C_j)\}$ with nondecreasing functions g_j and f_j , and $C_{\max} = \max\{C_j\}$. Here and below we assume that each maximum or summation is taken over all j . All data, decision variables, and values of the functions g_j and f_j are assumed to be nonnegative integers.

There are several approaches to attaining optimality in multicriterion optimization. In our paper, the criteria are independent; i.e., it is not required to minimize F_1 on the set of solutions minimizing F_2 and vice versa. In this case, the aim of the decision maker is to find a set of *nondominated solutions*. A solution is said to be nondominated if it outperforms any other solution on at least one criterion. A nondominated solution is also called a *Pareto optimal solution*. We note that there is no unique nondominated solution for our problem. A solution that performs well on one criterion may perform poorly on the other criterion. Indeed, if the jobs get fewer units of the resource, then a resource consumption, i.e., criterion F_1 , decreases. However, the job processing times p_j increase in this case leading to increasing of job completion times C_j and, consequently, criterion F_2 .

We now give formal definitions for the Pareto optimal solution, the *Pareto set*, and the *Pareto set ϵ -approximation*. Let S be the set of all feasible solutions to our problem.

DEFINITION 1.1. A feasible solution $s \in S$ is Pareto optimal if there is no feasible solution $q \in S$ such that $F_1(q) \leq F_1(s)$ and $F_2(q) \leq F_2(s)$, where at least one of the inequalities is strict.

DEFINITION 1.2. The Pareto set P is a set of Pareto optimal solutions such that there are no two solutions $s, q \in P$ with values $F_1(s) = F_1(q)$ and $F_2(s) = F_2(q)$.

DEFINITION 1.3. *Given $\epsilon > 0$, the Pareto set ϵ -approximation P_ϵ is a set such that for any Pareto optimal solution $s \in P$, there is a solution $q \in P_\epsilon$ satisfying*

$$F_1(q) \leq (1 + \epsilon)F_1(s) \text{ and } F_2(q) \leq (1 + \epsilon)F_2(s).$$

The paper is organized as follows. In the next section, we describe general schemes for the construction of the Pareto set and the Pareto set ϵ -approximation. In each iteration of these schemes, a Pareto optimal solution $s \in P$ and a solution $q \in P_\epsilon$, respectively, are found. An application of these schemes implies the existence of algorithms for solving the problems of minimizing F_1 subject to $F_2 \leq K$ and minimizing F_2 subject to $F_1 \leq K$, where K is a given number. In the following section, we provide computational complexity classification of various special cases of latter problems. In the fourth section, we present several dynamic programming formulations and approximation algorithms for the problems with $F_1 = \sum v_j x_j$ and $F_2 \in \{f_{max}, \sum w_j U_j\}$. We derive a new dynamic rounding technique to develop $(1 + \epsilon)$ -approximation algorithms. This technique not only rounds the problem parameters, as it is usually done in rounded dynamic programming (see Sahni (1977), Lawler (1982), Hansen (1980), Gens and Levner (1981), etc.), it also modifies the corresponding dynamic program.

2. General schemes. In this section, we describe general schemes for the construction of the Pareto set P and the Pareto set ϵ -approximation P_ϵ for our general problem.

It is convenient to adopt the three field notation of Graham et al. (1979) to denote our type of problems. In this notation, $1/\beta/\gamma$, the first field denotes the single machine processing system. The second field, $\beta \in \{b_j = b, a_j = a, \tau_j = \tau, d_j = d\}$, specifies some job characteristics (equal b_j, a_j, τ_j , or d_j , respectively). The third field is $\gamma \in \{(F_1, F_2), (F_1 \leq K, F_2), (F_1, F_2 \leq K)\}$, where (F_1, F_2) indicates the problem of finding the Pareto set, while $(F_1 \leq K, F_2)$ and $(F_1, F_2 \leq K)$ indicate the problem of minimizing F_2 subject to $F_1 \leq K$ and the problem of minimizing F_1 subject to $F_2 \leq K$, respectively. Our general problem is represented by $1/(F_1, F_2)$.

Let s_1 and s_2 be optimal solutions for the criteria F_1 and F_2 , respectively:

$$F_1(s_1) = \min\{F_1(s) | s \in S\} \text{ and } F_2(s_2) = \min\{F_2(s) | s \in S\}.$$

It is apparent that for each Pareto optimal solution $s \in P$ we have

$$F_1(s_1) \leq F_1(s) \leq F_1(s_2) \text{ and } F_2(s_2) \leq F_2(s) \leq F_2(s_1).$$

We now present a straightforward algorithm B for the construction of the Pareto set P for the general problem $1/(F_1, F_2)$. Set $K_1 = F_2(s_1)$. In each iteration $i = 1, \dots, l$ of this algorithm, we first solve the problem $1/(F_1, F_2 \leq K_i)$. Let $F_1^{(i)}$ be the minimal solution value for this problem. Then we solve the problem $1/(F_1 \leq F_1^{(i)}, F_2)$. If $s^{(i)}$ is an optimal solution to the latter problem, then $s^{(i)} \in P$. We set $K_{i+1} = F_2(s^{(i)}) - 1$ and go to the next iteration. Algorithm B is terminated when $F_1(s^{(i)}) = F_1(s_2)$. A formal description of Algorithm B is given below.

ALGORITHM B .

Step 1. Compute $F_2(s_1)$ and $F_1(s_2)$. Set $P = \emptyset$, $i = 1$, and $K_i = F_2(s_1)$.

Step 2. Find the minimal solution value $F_1^{(i)}$ to the problem $1/(F_1, F_2 \leq K_i)$. Find the optimal solution $s^{(i)}$ to the problem $1/(F_1 \leq F_1^{(i)}, F_2)$. Set $P = P \cup s^{(i)}$. If $F_1^{(i)} = F_1(s_2)$, then stop: the Pareto set P is constructed. Otherwise, set $K_{i+1} = F_2(s^{(i)}) - 1$, $i = i + 1$, and repeat Step 2.

THEOREM 2.1. *Algorithm B constructs the Pareto set P for the problem $1/(F_1, F_2)$ in $O(|P|(T_1 + T_2))$ time, assuming that the problems $1/(F_1, F_2 \leq K)$ and $1/(F_1 \leq K, F_2)$ are solved in $O(T_1)$ and $O(T_2)$ time, respectively, for any K .*

Proof. It is evident that $s^{(1)} \in P$ and, for every $s \in P - \{s^{(1)}\}$, we have $F_2(s) < F_2(s^{(1)})$. Our inductive assumption is that in each iteration i of the algorithm B we have $s^{(i)} \in P$ and inequality $F_2(s) < F_2(s^{(i)})$ is satisfied for every $s \in P - \{s^{(1)}, \dots, s^{(i)}\}$. If this assumption is correct for all i used in the algorithm, then the algorithm is also correct. Indeed, it follows from the description of the algorithm that $F_2(s^{(1)}) > F_2(s^{(2)}) > \dots > F_2(s^{(i)})$. Furthermore, according to our assumption we have $F_2(s^{(i)}) > F_2(s)$ for every $s \in P - \{s^{(1)}, \dots, s^{(i)}\}$. We then deduce that $s^{(i+1)} \neq s^{(j)}$, $j = 1, \dots, i$; i.e., a solution found in iteration $i + 1$ is a new element from P . Since $F_2(s) \geq F_2(s_2)$ for all $s \in P$ and algorithm B is terminated when a solution with value $F_2(s_2)$ is found, all elements $s \in P$ will be found and no element $s' \notin P$ can be found by Algorithm B .

Let our inductive assumption be satisfied for $j = i$. We show that it is also satisfied for $j = i + 1$.

Consider the problem $1/(F_1, F_2 \leq K_{i+1})$, where $K_{i+1} = F_2(s^{(i)}) - 1$. Due to the integrality of all parameters, $F_2 \leq K_{i+1}$ is equivalent to $F_2 < F_2(s^{(i)})$. Since $F_1^{(i+1)}$ is the minimal solution value for this problem, there is no feasible solution $s \in S$, $F_2(s) < F_2(s^{(i)})$ such that $F_1(s) < F_1^{(i+1)} = F_1(s^{(i+1)})$. Moreover, $s^{(i+1)}$ minimizes F_2 subject to $F_1(s) = F_1(s^{(i+1)})$. Therefore, due to the definition of the Pareto set P , there is only one solution $s \in P$ satisfying $F_2(s^{(i+1)}) \leq F_2(s) < F_2(s^{(i)})$. Clearly, such a solution can be $s^{(i+1)}$. Thus, $s^{(i+1)} \in P$ and for every $s \in P - \{s^{(1)}, \dots, s^{(i+1)}\}$ we have $F_2(s) < F_2(s^{(i+1)})$.

We now establish the time complexity of Algorithm B . In each iteration of Step 2, one new solution $s \in P$ is found. Therefore, the number of these iterations is exactly $|P|$. Each iteration requires $O(T_1 + T_2)$ time. Thus, Step 2 requires $O(|P|(T_1 + T_2))$ time, which is the overall time complexity of Algorithm B as well. $\square$

We note that the criteria can be switched when constructing Algorithm B . Also, for the bicriterion problem $1/(F_1, F_2)$, at least $|P|$ operations are required to obtain a solution. Therefore, if problems $1/(F_1, F_2 \leq K)$ and $1/(F_1 \leq K, F_2)$ are polynomially solvable, then Algorithm B is efficient even if $|P|$ is not polynomial in the problem instance length.

We now present an algorithm for the construction of the Pareto set ϵ -approximation P_ϵ for the problem $1/(F_1, F_2)$. Assume that lower and upper bounds for the values $F_1(s_1)$, $F_1(s_2)$, $F_2(s_1)$, and $F_2(s_2)$ are known such that

$$0 < L_1 \leq F_1(s_1) \leq F_1(s_2) \leq U_1 \text{ and } 0 < L_2 \leq F_2(s_2) \leq F_2(s_1) \leq U_2.$$

It is apparent that for each Pareto optimal solution $s \in P$, we have

$$(1) \quad L_1 \leq F_1(s) \leq U_1 \text{ and } L_2 \leq F_2(s) \leq U_2.$$

In our Algorithm B_ϵ for finding the Pareto set ϵ -approximation P_ϵ , the interval $[L_2, U_2]$ is divided into subintervals by the points $a_l, l = 0, 1, \dots, k$, so that $a_l = (1 + \epsilon/2)^l L_2$ for $l = 0, 1, \dots, k - 1$ and $a_k = U_2$. The number of these points k is defined so that $a_k \leq (1 + \epsilon/2)a_{k-1}$; i.e., $k \leq 1 + \log_{(1+\epsilon/2)}(U_2/L_2)$.

For each $l, l = 1, \dots, k$, a procedure $B(l)$ is applied. It is assumed that $B(l)$ has the following property. If there exists a Pareto optimal solution $s \in P$ with a value $F_2(s) \leq a_l$, then $B(l)$ finds a solution $s^{(l)} \in S$ such that

$$(2) \quad F_2(s^{(l)}) \leq a_l + \epsilon a_{l-1}/2, F_1(s^{(l)}) \leq (1 + \epsilon) \min\{F_1(q) | q \in P, F_2(q) \leq a_l\}.$$

The set P_ϵ is the set of all solutions $s^{(l)}$ found by $B(l)$ for $l = 1, \dots, k$. Below we present an approach to constructing the procedure $B(l)$. We now give a formal description of the algorithm B_ϵ , assuming that the procedure $B(l)$ is determined.

ALGORITHM B_ϵ .

Step 1. Set $P_\epsilon = \emptyset$, $a_0 = L_2$, and $l = 1$.

Step 2. If $(1 + \epsilon/2)a_{l-1} < U_2$, then set $a_l = (1 + \epsilon/2)a_{l-1}$; otherwise, set $a_l = U_2$. Apply the procedure $B(l)$. If it finds a solution $s^{(l)} \in S$ satisfying (2), then set $P_\epsilon = P_\epsilon \cup s^{(l)}$. If $(1 + \epsilon/2)a_{l-1} < U_2$, then set $l = l + 1$ and repeat Step 2; otherwise, stop: the Pareto set ϵ -approximation P_ϵ is constructed.

THEOREM 2.2. *Algorithm B_ϵ constructs the Pareto set ϵ -approximation for the problem $1/(F_1, F_2)$ in $O(T \log_{(1+\epsilon/2)}(U_2/L_2))$ time, assuming that for $l = 1, \dots, k$, the procedure $B(l)$ finds a solution $s^{(l)} \in S$ satisfying (2) in $O(T)$ time if there exists a Pareto optimal solution s with a value $F_2(s) \leq a_l$.*

Proof. Consider any Pareto optimal solution $s \in P$. Due to the inequalities (1), there exists a number l , $1 \leq l \leq k$, such that $a_{l-1} \leq F_2(s) \leq a_l$. Hence, the procedure $B(l)$ finds a solution $s^{(l)} \in S$ such that

$$F_2(s^{(l)}) \leq a_l + \epsilon a_{l-1}/2 \leq (1 + \epsilon)a_{l-1} \leq (1 + \epsilon)F_2(s),$$

$$F_1(s^{(l)}) \leq (1 + \epsilon) \min\{F_1(q) | q \in P, F_2(q) \leq a_l\} \leq (1 + \epsilon)F_1(s).$$

Thus, the Pareto set ϵ -approximation P_ϵ is constructed by Algorithm B_ϵ .

The number of iterations of Step 2 is at most $1 + \lceil \log_{(1+\epsilon/2)}(U_2/L_2) \rceil$, and each iteration requires $O(T)$ time. Therefore, the time complexity of the algorithm B_ϵ is $O(T \log_{(1+\epsilon/2)}(U_2/L_2))$. $\square$

We now give a definition of a (ϵ, ρ) -approximation algorithm for the problems $1/(F_1, F_2 \leq K)$ and $1/(F_1 \leq K, F_2)$. Let F_1^* and F_2^* be the optimal solution values for these problems, respectively.

DEFINITION 2.3. *An approximation algorithm for the problem $1/(F_1, F_2 \leq K)$ (problem $1/(F_1 \leq K, F_2)$) is called a (ϵ, ρ) -approximation algorithm if, for any $\epsilon > 0, \rho > 0$ and an arbitrary problem instance, it delivers a solution with values $F_2 \leq (1 + \rho)K$ ($F_1 \leq (1 + \rho)K$) and $F_1 \leq (1 + \epsilon)F_1^*$ ($F_2 \leq (1 + \epsilon)F_2^*$).*

We note that a (ϵ, ρ) -approximation algorithm for the problem $1/(F_1, F_2 \leq a_l)$ can be used as the procedure $B(l)$ in the algorithm B_ϵ if $\rho \leq \epsilon/(2 + \epsilon)$. Indeed, if s is a solution delivered by this algorithm, then, using inequality $a_l \leq (1 + \epsilon/2)a_{l-1}$ and Definition 2.3, we have $F_2(s) \leq (1 + \rho)a_l \leq (1 + \epsilon/(2 + \epsilon))a_l \leq a_l + \epsilon a_{l-1}/2$ and $F_1(s) \leq (1 + \epsilon) \min\{F_1(q) | q \in S, F_2(q) \leq a_l\} \leq (1 + \epsilon) \min\{F_1(q) | q \in P, F_2(q) \leq a_l\}$; i.e., (2) is satisfied. In the following section, we give an example of a $(\epsilon, 0)$ -approximation algorithm for the problem $1/(\sum v_j x_j, f_{\max} \leq K)$ and a (ϵ, ρ) -approximation algorithm for the problem $1/(\sum v_j x_j \leq K, \sum w_j U_j)$.

3. Computational complexity. In this section, we study the computational complexities of various special cases of the problems $1/(F_1, F_2 \leq K)$ and $1/(F_1 \leq K, F_2)$.

3.1. Problems with $F_1 = g_{\max}$. We first note that for the problem $1/(g_{\max} \leq K, F_2)$, we can define the optimal resource allocation $(x_1^*, \dots, x_n^*)$ as follows. The inequality $\max\{g_j(x_j)\} \leq K$ is satisfied if and only if $x_j \leq g_j^{-1}(K)$ for all j , where $g_j(g_j^{-1}(K)) \leq K$ and $g_j(g_j^{-1}(K) + 1) > K$. Clearly, there exists an optimal solution

to the problem $1/(g_{max} \leq K, F_2)$ in which $x_j^* = \min\{g_j^{-1}(K), \tau_j\}$ for $j = 1, \dots, n$. Hence, this problem reduces to one of finding a sequence of jobs with externally given processing times $p_j = b_j - a_j x_j^*$ to minimize F_2 . If $F_2 = \max\{f(C_j - d_j)\}$, where f is an arbitrary nondecreasing function, then the earliest due date (EDD) sequence is optimal, where jobs are sequenced in nondecreasing order of their due dates. If $F_2 = \sum w_j C_j$, then the shortest weighted processing time (SWPT) sequence is optimal, where jobs are sequenced in nondecreasing order of the values p_j/w_j . If $F_2 = f_{max}$ or $F_2 = \sum U_j$, then an optimal sequence can be found in $O(n \log n)$ time using Lawler's (1973) algorithm or Moore's (1968) algorithm, respectively. As for the problem $1/(g_{max} \leq K, \sum w_j U_j)$, an evident transformation from the NP-complete problem PARTITION (Garey and Johnson (1979)) shows that the decision version of this problem is NP-complete. Thus, the following theorem holds.

THEOREM 3.1. *The problem $1/(g_{max} \leq K, F_2)$ is solved in $O(n \log n)$ time for $F_2 \in \{f_{max}, \sum U_j, \sum w_j C_j\}$ and NP-hard for $F_2 = \sum w_j U_j$.*

We now show that the problem $1/(g_{max}, F_2 \leq K)$ can also be solved in polynomial time for any $F_2 \in \{f_{max}, \sum U_j, \sum w_j C_j\}$.

Let $F_2(x)$ be the minimal solution value for the criterion F_2 subject to the resource allocation $x = (x_1, \dots, x_n)$. As it is shown above, the value of $F_2(x)$ and the corresponding job sequence can be found in $O(n \log n)$ time for any x and $F_2 \in \{f_{max}, \sum U_j, \sum w_j C_j\}$. Define $0 = (0, \dots, 0)$ and $\tau = (\tau_1, \dots, \tau_n)$. If $F_2(0) \leq K$, then $(0, \dots, 0)$ is an optimal resource allocation to the problem $1/(g_{max}, F_2 \leq K)$. If $F_2(\tau) > K$, then there is no solution to this problem. Assume that $F_2(0) > K$ and $F_2(\tau) \leq K$. Define $G(x) = \max\{g_j(x_j)\}$, and perform a bisection search in the range $G(0), G(0) + 1, \dots, G(\tau)$ as follows. Set $L = G(0) \geq 0$ and $R = G(\tau)$. In each iteration of our search, we calculate $M = (L + R)/2$ and find the maximal values x_j^M , $j = 1, \dots, n$, for which $g_{max} \leq M$ is satisfied. As is shown above, $x_j^M = \min\{g_j^{-1}(M), \tau_j\}$ for $j = 1, \dots, n$. If $F_2(x^M) > K$, set $L = M$; if $F_2(x^M) \leq K$, set $R = M$, then go to the next iteration. The procedure is terminated when $R - L < 1$. In this case, x^R is an optimal resource allocation to the problem $1/(g_{max}, F_2 \leq K)$. Note that the corresponding optimal job sequence is already found. The number of iterations of the above procedure is no greater than $\log G(\tau)$. Thus, we have the following theorem.

THEOREM 3.2. *For $F_2 \in \{f_{max}, \sum U_j, \sum w_j C_j\}$, the problem $1/(g_{max}, F_2 \leq K)$ is solved in $O(n \log n \log(\max\{g_j(\tau_j)\}))$ time.*

It should be noted that the above bisection search procedure can be generalized to solve an arbitrary problem $1/\beta/(F_1, F_2 \leq K)$ if there is an algorithm for the problem $1/\beta/(F_1 \leq K, F_2)$. This generalized bisection search procedure *BS* is as follows.

PROCEDURE BS.

- Step 1.** Define lower and upper bounds for the minimal solution value F_1^* to the criterion F_1 : $L \leq F_1^* \leq R$. Let $F_2(M)$ be the minimal solution value to the problem $1/\beta/(F_1 \leq M, F_2)$. If $F_2(L) \leq K$, then stop: a solution to the problem $1/\beta/(F_1 \leq L, F_2)$ is a solution to the problem $1/\beta/(F_1, F_2 \leq K)$. If $F_2(R) > K$, then stop: there is no solution to the latter problem. If $F_2(L) > K$ and $F_2(R) \leq K$, then set $E = L, H = R$ and go to Step 2.
- Step 2.** If $H - E < 1$, then stop: a solution to the problem $1/\beta/(F_1 \leq H, F_2)$ is a solution to the problem $1/\beta/(F_1, F_2 \leq K)$. Otherwise, calculate $M = (E + H)/2$ and solve the problem $1/\beta/(F_1 \leq M, F_2)$. If $F_2(M) > K$, set $E = M$. If $F_2(M) \leq K$, set $H = M$. In either case, repeat Step 2.

It is evident that the number of iterations of Step 2 does not exceed $\log(R - L)$. If the problem $1/\beta/(F_1 \leq K, F_2)$ is solved in $O(T)$ time, then Procedure *BS* runs in $O(T \log(R - L))$ time.

Since the criteria F_1 and F_2 are independent, the same procedure can be employed to solve the problem $1/\beta/(F_1 \leq K, F_2)$ if there is an algorithm for the problem $1/\beta/(F_1, F_2 \leq K)$. Let F_2^* be the minimal solution value to the problem $1/\beta/(F_1 \leq K, F_2)$. Then, the following theorem holds.

THEOREM 3.3. *If the problem $1/\beta/(F_1 \leq K, F_2)$ (problem $1/\beta/(F_1, F_2 \leq K)$) can be solved in $O(T)$ time and $L \leq F_1^* \leq R$ ($L \leq F_2^* \leq R$), then the problem $1/\beta/(F_1, F_2 \leq K)$ (problem $1/\beta/(F_1 \leq K, F_2)$) can be solved in $O(T \log(R - L))$ time using Procedure *BS*.*

3.2. Problems with $F_1 = \sum v_j x_j$. It is evident that the problem $1/(\sum v_j x_j, f_{\max} \leq K)$ is equivalent to one of minimizing $\sum v_j x_j$ subject to each job j meeting the deadline $f_j^{-1}(K)$, which is defined in the same way as $g_j^{-1}(K)$. The latter problem has been studied by Janiak and Kovalyov (1993). It follows from their research that the problem $1/b_j = b, \tau_j = 1/(\sum v_j x_j, C_{\max} \leq K)$ is NP-hard and the problems $1/a_j = a/(\sum v_j x_j, f_{\max} \leq K)$ and $1/(\sum x_j, f_{\max} \leq K)$ are both solvable in $O(n \log n)$ time by a modification of Moore's (1968) algorithm.

Since for the minimal solution value $f_{\max}^*$ for the criterion $f_{\max}$ we have $0 \leq f_{\max}^* \leq \max\{f_j(\sum_{i=1}^n b_i)\}$, Theorem 3.3 shows that the problems $1/a_j = a/(\sum v_j x_j \leq K, f_{\max})$ and $1/(\sum x_j \leq K, f_{\max})$ can both be solved in $O(n \log n \log(\max\{f_j(\sum_{i=1}^n b_i)\}))$ time by applying Procedure *BS*.

The problem with $F_1 = \sum v_j x_j$ and $F_2 = \sum w_j U_j$ has been studied by Cheng, Chen, and Li (1996). They proved that the problem $1/a_j = 1, d_j = d/(\sum x_j, \sum U_j \leq K)$ is NP-hard.

We now begin to study the problem with $F_1 = \sum v_j x_j$ and $F_2 = \sum w_j C_j$. As far as we know, this problem has not been considered in the literature.

THEOREM 3.4. *The problem $1/\tau_j = 1/(\sum v_j x_j \leq K, \sum w_j C_j)$ is NP-hard.*

Proof. We show that the decision version of the above problem is NP-complete by a transformation from the NP-complete problem PARTITION (Garey and Johnson (1979)). Given positive integers $r_1, \dots, r_n$, is there a set $Q \subseteq N = \{1, \dots, n\}$ such that $\sum_{j \in Q} r_j = R$, where $\sum_{j \in N} r_j = 2R$? Given any instance of PARTITION, we construct an instance of our problem in which there are $n+1$ jobs with $v_j = b_j = a_j = r_j$, $w_j = 1$ for $j \in N$, $b_{n+1} = a_{n+1} = n^3 R^2$, $w_{n+1} = n^2 R$, $v_{n+1} = R + 1$, and $x_j \in \{0, 1\}$ for all j . We set $K = R$ and show that there exists a set $Q \subseteq N$ for which $\sum_{j \in Q} r_j = R$ if and only if there exists a solution to our problem for which $\sum_{j=1}^{n+1} v_j x_j \leq K$ and $\sum_{j=1}^{n+1} w_j C_j \leq L = nR + n^2 R(R + n^3 R^2)$.

If there is a set $Q \subseteq N$ for which $\sum_{j \in Q} r_j = R$, then we allocate the resource so that $x_j = 1$, $j \in Q$, $x_j = 0$, $j \in N - Q$, and $x_{n+1} = 0$. We assign job $n+1$ to be scheduled last, jobs $j \in Q$ to be scheduled first in an arbitrary order, and jobs $j \in N - Q$ to be scheduled after the last job $j \in Q$ in an arbitrary order. For this solution, we have $\sum_{j=1}^{n+1} v_j x_j = \sum_{j \in Q} v_j = R = K$ and $\sum_{j=1}^{n+1} w_j C_j = \sum_{j \in N-Q} w_j C_j + w_{n+1} C_{n+1} = \sum_{j \in N-Q} C_j + n^2 R(\sum_{j \in N-Q} b_j + n^3 R^2) \leq nR + n^2 R(R + n^3 R^2) = L$.

Conversely, suppose there is a solution to our problem for which $\sum_{j=1}^{n+1} v_j x_j \leq K$ and $\sum_{j=1}^{n+1} w_j C_j \leq L$. We note that $x_{n+1} = 0$, since otherwise $\sum_{j=1}^{n+1} v_j x_j \geq v_{n+1} = K + 1$. Therefore, we have $\sum_{j \in N} v_j x_j \leq R$ and $\sum_{j \in N, x_j=0} r_j \geq R$. Furthermore, due

to Smith's (1956) rule, there exists a schedule for which $\sum_{j=1}^{n+1} w_j C_j \leq L$ and jobs are sequenced in nondecreasing order of the values $(b_j - a_j x_j)/w_j$. We have

$$(b_j - a_j x_j)/w_j = \begin{cases} 0 & \text{if } x_j = 1, \\ r_j & \text{if } j \in N, x_j = 0, \\ nR & \text{if } j = n+1. \end{cases}$$

Thus, we can assume that jobs with $x_j = 1$ are scheduled first, then jobs with $x_j = 0$ are scheduled, and job $n+1$ is scheduled last. For this schedule we have

$$\begin{aligned} L = nR + n^2 R(R + n^3 R^2) &\geq \sum_{j=1}^{n+1} w_j C_j = \sum_{j \in N, x_j=0} C_j + w_{n+1} C_{n+1} \\ &\geq \sum_{j \in N, x_j=0} r_j + n^2 R \left(\sum_{j \in N, x_j=0} r_j + n^3 R^2 \right), \end{aligned}$$

whence it follows that $\sum_{j \in N, x_j=0} r_j \leq R$. We deduce that $\sum_{j \in N, x_j=0} r_j = R$. Therefore, PARTITION has a solution. $\square$

We now derive a polynomial-time algorithm for the problem $1/a_j = a, b_j = b/(\sum x_j \leq K, \sum C_j)$. We first note that, for any resource allocation $(x_1, \dots, x_n)$, it is optimal to sequence the jobs in the shortest processing time (SPT) order so as $b - ax_{i_1} \leq b - ax_{i_2} \leq \dots \leq b - ax_{i_n}$; i.e., $x_{i_1} \geq x_{i_2} \geq \dots \geq x_{i_n}$. Since the jobs differ only with the values τ_j , we deduce that the sequence of jobs in nonincreasing order of τ_j is optimal.

Number the jobs so that $\tau_1 \geq \dots \geq \tau_n$.

Set $t = \min\{K, \tau_1\}$. We show that in any optimal solution $x_1 = t$. Assume $t - x_1 = \delta > 0$. Note that $\sum_{j=1}^n x_j = K$. If $\sum_{j=1}^n x_j < K$, then the value of the objective function can be decreased by allocating $K - \sum_{j=1}^n x_j$ additional units of the resource. Thus, $\sum_{j=2}^n x_j = K - x_1 = K - t + \delta \geq \delta$. Move δ units of the resource from $x_2, \dots, x_n$ to x_1 . For the new solution, we have $x_1 = t$ and the objective function value is decreased. Therefore, $x_1 = \min\{K, \tau_1\}$ in any optimal solution and the original problem reduces to one of minimizing $\sum_{j=2}^n C_j$ subject to $\sum_{j=2}^n x_j \leq K_1$, where $K_1 = K - x_1$. Recursively, the latter problem reduces to one of minimizing $\sum_{j=3}^n C_j$ subject to $\sum_{j=3}^n x_j \leq K_2$, where $K_2 = K_1 - x_2$, $x_2 = \min\{K_1, \tau_2\}$, and so on.

We now describe Algorithm G , in which the jobs are assigned to the end of the current sequence in nonincreasing order of the values τ_j and each current job gets as much allocation of the resource as possible. Thus, Algorithm G is a *greedy* algorithm. A formal description of this algorithm is as follows.

ALGORITHM G .

Step 1. Number jobs so that $\tau_1 \geq \dots \geq \tau_n$. Set $j = 1$.

Step 2. Compute $x_j = \min\{K, \tau_j\}$. If $j = n$, then stop: the job sequence $(1, \dots, n)$ and the resource allocation $(x_1, \dots, x_n)$ constitute an optimal solution. Otherwise, set $K = K - x_j$, $j = j + 1$ and repeat Step 2.

THEOREM 3.5. *Algorithm G solves the problem $1/a_j = a, b_j = b/(\sum x_j \leq K, \sum C_j)$ in $O(n \log n)$ time.*

Theorems 3.3 and 3.5 show that the problem $1/a_j = a, b_j = b/(\sum x_j, \sum C_j \leq K)$ can be solved in $O(n \log n \log(\sum \tau_j))$ time by applying Procedure BS .

TABLE 1
Complexities of the problems $1/(F_1, F_2 \leq K)$ and $1/(F_1 \leq K, F_2)$.

Problem	Complexity	Reference
$1/(g_{max} \leq K, F_2),$ $F_2 \in \{f_{max}, \sum U_j, \sum w_j C_j\}$	$n \log n$	Theorem 3.1
$1/(g_{max}, F_2 \leq K),$ $F_2 \in \{f_{max}, \sum U_j, \sum w_j C_j\}$	$n \log n \log(\max\{g_j(\tau_j)\})$	Theorem 3.2
$1/(g_{max} \leq K, \sum w_j U_j)$	NP-hard	Theorem 3.1
$1/(g_{max}, \sum w_j U_j \leq K)$	NP-hard	
$1/b_j = b, \tau_j = 1/(\sum v_j x_j, C_{max} \leq K)$	NP-hard	Janiak and Kovalyov (1993)
$1/b_j = b, \tau_j = 1/(\sum v_j x_j \leq K, C_{max})$	NP-hard	Janiak and Kovalyov (1993)
$1/a_j = a/(\sum v_j x_j, f_{max} \leq K)$	$n \log n$	and this section
$1/a_j = a/(\sum v_j x_j \leq K, f_{max})$	$n \log n \log(\max\{f_j(\sum_{i=1}^n b_i)\})$	Janiak and Kovalyov (1993)
$1/(\sum x_j, f_{max} \leq K)$	$n \log n$	and Theorem 3.3
$1/(\sum x_j \leq K, f_{max})$	$n \log n \log(\max\{f_j(\sum_{i=1}^n b_i)\})$	Janiak and Kovalyov (1993)
$1/a_j = 1, d_j = d/(\sum x_j, \sum U_j \leq K)$	NP-hard	and Theorem 3.3
$1/a_j = 1, d_j = d/(\sum x_j \leq K, \sum U_j)$	NP-hard	Cheng, Chen, and Li (1994)
$1/\tau_j = 1/(\sum v_j x_j \leq K, \sum w_j C_j)$	NP-hard	Theorem 3.4
$1/\tau_j = 1/(\sum v_j x_j, \sum w_j C_j \leq K)$	NP-hard	
$1/a_j = a, b_j = b/(\sum x_j \leq K, \sum C_j)$	$n \log n$	Theorem 3.5
$1/a_j = a, b_j = b/(\sum x_j, \sum C_j \leq K)$	$n \log n \log(\sum \tau_j)$	Theorem 3.3

Finally, the computational complexities of various special cases of the problems $1/(F_1, F_2 \leq K)$ and $1/(F_1 \leq K, F_2)$ are given in Table 1. Note that the NP-hardness results for several problems presented in this table are established using the following evident statement (see also Lee and Vairaktarakis (1993)).

THEOREM 3.6. *If the decision version of one of the problems $1/\beta/(F_1, F_2 \leq K)$ and $1/\beta/(F_1 \leq K, F_2)$ is NP-complete, then both problems are NP-hard.*

The complexities of all other special cases of the problems $1/(F_1, F_2 \leq K)$ and $1/(F_1 \leq K, F_2)$ which are not covered by those presented in Table 1 are still unknown. The most interesting open questions are the complexities of the special cases with $F_1 = \sum v_j x_j$ and $F_2 = \sum w_j C_j$.

As noted in the previous section, if both problems $1/\beta/(F_1, F_2 \leq K)$ and $1/\beta/(F_1 \leq K, F_2)$ are polynomially solvable, then the problem $1/\beta/(F_1, F_2)$ can be efficiently solved using Algorithm B. Thus, the problem $1/(g_{max}, F_2)$ can be solved in $O(|P|n(\log n + \log(\max\{g_j(\tau_j)\})))$ time for $F_2 \in \{f_{max}, \sum U_j, \sum w_j C_j\}$, both problems $1/a_j = a/(\sum v_j x_j, f_{max})$ and $1/(\sum x_j, f_{max})$ in $O(|P|n(\log n + \log(\max\{f_j(\sum_{i=1}^n b_i)\})))$ time, and the problem $1/a_j = a, b_j = b/(\sum x_j, \sum C_j)$ in $O(|P|n(\log n + \log(\sum \tau_j)))$ time.

4. Dynamic programming and approximation. The time complexities of Algorithms B and B_ϵ presented in section 2 show that these algorithms are efficient for the problem $1/(F_1, F_2)$ even if the problems $1/(F_1, F_2 \leq K)$ and $1/(F_1 \leq K, F_2)$ are NP-hard but there are pseudopolynomial algorithms or (ϵ, ρ) -approximation algorithms to solve them. In this section, we give several examples of such algorithms

for the problems with $F_1 = \sum v_j x_j$ and $F_2 \in \{f_{max}, \sum w_j U_j\}$. We first consider the problem $1/(\sum v_j x_j, f_{max})$.

For the problem to minimize $\sum v_j x_j$ subject to each job j meeting the deadline d_j , a dynamic programming algorithm and a $(\epsilon, 0)$ -approximation algorithm are presented by Janiak and Kovalyov (1993). The time complexities of these algorithms are $O(D)$ and $O(E)$, respectively, where $D = n(\sum w_j \tau_j)^2$ and $E = n^3/\epsilon^2 + n^3 \log n + n \log(\max\{w_j \tau_j\})$. As it is shown in the previous section, these algorithms can be applied to solve the problem $1/(\sum v_j x_j, f_{max} \leq K)$ if we set $d_j = f_j^{-1}(K)$ for $j = 1, \dots, n$. Furthermore, Theorem 3.3 shows that the problem $1/(\sum v_j x_j \leq K, f_{max})$ can be solved in $O(D \log(\max\{f_j(\sum_{i=1}^n b_i)\}))$ time by applying Procedure *BS*. Then, since $0 \leq f_{max} \leq \max\{f_j(\sum_{i=1}^n b_i)\}$ for the value f_{max} of any feasible solution, Theorems 2.1 and 2.2 show that, for the problem $1/(\sum v_j x_j, f_{max})$, the Pareto set P can be constructed in $O(|P|D \log(\max\{f_j(\sum_{i=1}^n b_i)\}))$ time by applying Algorithm *B* and the Pareto set ϵ -approximation P_ϵ can be constructed in $O(E \log_{(1+\epsilon/2)}(\max\{f_j(\sum_{i=1}^n b_i)\}))$ time by applying Algorithm B_ϵ .

We now present dynamic programming algorithms for the problems $1/(\sum v_j x_j \leq K, \sum w_j U_j)$ and $1/(\sum v_j x_j, \sum w_j U_j \leq K)$. We note that dynamic programming algorithms for these problems are already constructed by Cheng et al. (1998). However, in our algorithms, different definitions of the function values and state variables are used and so a comparison of the time complexities of our algorithms with those of Cheng et al. is not possible. Also, we show that our dynamic programming algorithms can be transformed into (ϵ, ρ) -approximation algorithms that are unlikely for the algorithms presented in Cheng et al. (1998).

It is convenient to introduce some terminology. Given a solution to the problem $1/(\sum v_j x_j, \sum w_j U_j)$, job j is *late* if it is completed after the due date d_j : $C_j > d_j$; otherwise, it is *early*. Our algorithms as well as the algorithms presented in Cheng and Chen (1994) are based on the following evident observation. There exists an optimal solution to each of the problems $1/(\sum v_j x_j \leq K, \sum w_j U_j)$ and $1/(\sum v_j x_j, \sum w_j U_j \leq K)$ with the following properties:

- Early jobs are sequenced in EDD order.
- Late jobs are sequenced in an arbitrary order after the last early job and, for each late job j , we have $x_j = 0$.

Assume that the jobs are numbered in EDD order so that $d_1 \leq \dots \leq d_n$. In our algorithms, jobs are considered in natural order $1, \dots, n$. The following two possible scheduling choices for each job j are considered:

- Job j is scheduled as the last early job if it will be completed before the due date d_j . In this case, $0 \leq x_j \leq \tau_j$, the completion time of the last early job is increased by $b_j - a_j x_j$ and the cost $v_j x_j$ is incurred in the first criterion.
- Job j is scheduled as a late job. In this case, $x_j = 0$ and the cost w_j is incurred in the second criterion.

Let W^* be the optimal solution value for the problem $1/(\sum v_j x_j \leq K, \sum w_j U_j)$, and let Y be a positive integer. We now describe our first dynamic programming Algorithm $D1(Y)$, which either solves the problem $1/(\sum v_j x_j \leq K, \sum w_j U_j)$ or establishes that $W^* > Y$. In this algorithm, the completion time of the last early job is a function value and the weighted number of late jobs and the total weighted resource consumption are state variables. More precisely, we recursively compute the value of $C_j(W, V)$, which represents the minimal completion time of the last early job subject to j jobs having been scheduled, the total weighted number of late jobs equal to W , and the total weighted resource consumption equal to V . A formal description of this dynamic programming algorithm is as follows.

ALGORITHM $D1(Y)$.

Step 1 (Initialization). Number jobs in EDD order so that $d_1 \leq \dots \leq d_n$. Set $C_j(W, V) = 0$ for $j = 0$, $W = 0$, and $V = 0$. Set $C_j(W, V) = \infty$ otherwise. Set $j = 1$.

Step 2 (Recursion). Compute the following for all $0 \leq W \leq Y$ and $0 \leq V \leq K$:

$$C_j(W, V) = \min \begin{cases} C_{j-1}(W - w_j, V), \\ \min\{T(x_j) | T(x_j) \leq d_j, x_j \in \{0, 1, \dots, \tau_j\}\}, \end{cases}$$

where

$$T(x_j) = C_{j-1}(W, V - v_j x_j) + b_j - a_j x_j.$$

If $j = n$, go to Step 3; otherwise, set $j = j + 1$ and repeat Step 2.

Step 3 (Optimal solution). If $C_n(W, V) = \infty$ for all $0 \leq W \leq Y$ and $0 \leq V \leq K$, then $W^* > Y$; otherwise, define

$$W^* = \min\{W | C_n(W, V) < \infty, 0 \leq W \leq Y, 0 \leq V \leq K\}$$

and use backtracking to find the corresponding optimal solution.

THEOREM 4.1. *Algorithm $D1(Y)$ solves the problem $1/(\sum v_j x_j \leq K, \sum w_j U_j)$ if and only if $W^* \leq Y$ and has $O(nYK \sum \tau_j)$ running time.*

Proof. Since all possible scheduling choices for each job j and all possible resource allocations x_j are considered in Step 2, the general dynamic programming justification for scheduling problems (Rothkopf (1966), Lawler and Moore (1969)) shows that in Step 2, all possible objective values $W \leq Y$ are obtained. In Step 3, if $C_n(W, V) = \infty$ for all $0 \leq W \leq Y$ and $0 \leq V \leq K$, then there is no solution with a value $W \leq Y$, i.e., $W^* > Y$, which proves the necessity. Alternatively, if $C_n(W, V) < \infty$ for some W and V , then W^* is the minimal objective value among those obtained in Step 2. Thus, the sufficiency is also proved.

Since there are n different values of j , $K + 1$ different values of V , and $Y + 1$ different values of W , Steps 1 and 3 require $O(nYK)$ operations. In each iteration of Step 2, the value of $T(x_j)$ is computed for $0 \leq V \leq K$ and $0 \leq W \leq Y$. Each calculation of $T(x_j)$ requires $O(\tau_j)$ operations. Thus, Step 2 can be performed in $O(nYK \sum \tau_j)$ time, which is the overall time complexity of $D1(Y)$ as well. $\square$

It is apparent that $D1(\sum w_j)$ is a pseudopolynomial algorithm for the problem $1/(\sum v_j x_j \leq K, \sum w_j U_j)$.

An analysis of Algorithm $D1(Y)$ shows that it can easily be modified to solve the problem $1/(\sum v_j x_j, \sum w_j U_j \leq K)$. Assume that X is a guess for an upper bound for the optimal objective value V^* of this problem: $V^* \leq X$. In Algorithm $D1(Y)$, we set $Y = K$ and $K = X$. Besides, in Step 3 of this algorithm, if $C_n(W, V) = \infty$ for all $0 \leq W \leq K$ and $0 \leq V \leq X$, then $V^* > X$; otherwise,

$$V^* = \min\{V | C_n(W, V) < \infty, 0 \leq W \leq K, 0 \leq V \leq X\}.$$

We denote this modified algorithm by $D2(X)$.

THEOREM 4.2. *Algorithm $D2(X)$ solves the problem $1/(\sum v_j x_j, \sum w_j U_j \leq K)$ if and only if $V^* \leq X$ and has $O(nXK \sum \tau_j)$ running time.*

We note that $D2(\sum v_j \tau_j)$ is a pseudopolynomial algorithm for the problem $1/(\sum v_j x_j, \sum w_j U_j \leq K)$. Moreover, an analysis of Algorithms $D1(Y)$ and $D2(X)$ shows that several special cases of the problems $1/(\sum x_j \leq K, \sum U_j)$ and $1/(\sum x_j, \sum U_j \leq K)$ in which all τ_j are constant or bounded by a polynomial in n can be solved

in polynomial time. We also note that, by incorporating Algorithms $D1(\sum w_j)$ and $D2(\sum v_j \tau_j)$ into Algorithm B , the Pareto set P for the problem $1/(\sum v_j x_j, \sum w_j U_j)$ can be found in $O(|P|nK \sum (w_j + v_j \tau_j) \sum \tau_j)$ time.

We now show how to construct the Pareto set ϵ -approximation for this problem. We first modify the algorithm $D1(Y)$ to be a (ϵ, ρ) -approximation algorithm for the problem $1/(\sum v_j x_j \leq K, \sum w_j U_j)$.

Assume that numbers L and R are known such that $0 < L \leq W^* \leq R$. We set $\mu = \epsilon L/n$, $\delta = \rho K/n$ and modify the algorithm $D1(Y)$ as follows. Define $r_j(l) = \max\{x_j | 0 \leq x_j \leq \tau_j, \lfloor v_j x_j / \delta \rfloor = l\}$ for $j = 1, \dots, n$ and $l = 0, 1, \dots, \lfloor K/\delta \rfloor$. Substitute $\lfloor R/\mu \rfloor$ for Y , $\lfloor w_j/\mu \rfloor$ for w_j , $\lfloor K/\delta \rfloor$ for K , $\lfloor v_j x_j / \delta \rfloor$ for $v_j x_j$, and $x_j \in \{r_j(0), r_j(1), \dots, r_j(\lfloor K/\delta \rfloor)\}$ for $x_j \in \{0, 1, \dots, \tau_j\}$ in the description of the algorithm $D1(Y)$. Denote the modified algorithm by $A_{\epsilon, \rho}(L, R)$. Our method to develop $A_{\epsilon, \rho}(L, R)$ differs from the known techniques “rounding” and “interval partitioning” (Sahni (1977)) and can be considered as a new *dynamic rounding technique* to develop $(1 + \epsilon)$ -approximation algorithms. We now establish the correctness and the running time of this algorithm.

THEOREM 4.3. *The algorithm $A_{\epsilon, \rho}(L, R)$ is a (ϵ, ρ) -approximation algorithm for the problem $1/(\sum v_j x_j \leq K, \sum w_j U_j)$ and has $O(n^5 R/(\epsilon \rho^2 L))$ running time.*

Proof. Assume that there exists an optimal solution to the problem $1/(\sum v_j x_j \leq K, \sum w_j U_j)$ with a resource allocation $(x_1^*, \dots, x_n^*)$, job completion times C_j^* , $j = 1, \dots, n$, and the objective function value of $W^* = \sum w_j U_j^* \leq R$. Set $x'_j = r_j(\lfloor v_j x_j^* / \delta \rfloor)$. By the definition of $r_j(l)$, we have $x'_j \geq x_j^*$ for $j = 1, \dots, n$. Therefore, a solution with the same job sequence as in the optimal solution and with a resource allocation $(x'_1, \dots, x'_n)$ will have job completion times $C'_j \leq C_j^*$ for $j = 1, \dots, n$. For this solution, define $U'_j = 0$ if $C'_j \leq d_j$ and $U'_j = 1$ otherwise. We have

$$\sum \lfloor w_j / \mu \rfloor U'_j \leq \sum \lfloor w_j / \mu \rfloor U_j^* \leq \lfloor R / \mu \rfloor,$$

$$\sum \lfloor v_j x'_j / \delta \rfloor = \sum \lfloor v_j x_j^* / \delta \rfloor \leq \lfloor K / \delta \rfloor.$$

Thus, in Step 3 of Algorithm $A_{\epsilon, \rho}(L, R)$, there will be at least one solution for which $C_n(W, V) < \infty$. Let $(x_1^0, \dots, x_n^0)$ and $(U_1^0, \dots, U_n^0)$ be a resource allocation and a sequence of values U_j , $j = 1, \dots, n$, respectively, given by $A_{\epsilon, \rho}(L, R)$. Making use of the inequalities $\lfloor y \rfloor \leq y < \lfloor y \rfloor + 1$, where y is any real number, we have

$$W^0 = \sum w_j U_j^0 \leq \mu \sum \lfloor w_j / \mu \rfloor U_j^0 + n\mu,$$

$$V^0 = \sum v_j x_j^0 \leq \delta \sum \lfloor v_j x_j^0 / \delta \rfloor + n\delta.$$

According to the description of $A_{\epsilon, \rho}(L, R)$, $\sum \lfloor v_j x_j^0 / \delta \rfloor \leq \lfloor K / \delta \rfloor$ holds. Hence, $V^0 \leq (1 + \rho)K$. Furthermore, since the sequence $(U_1^0, \dots, U_n^0)$ minimizes $\sum \lfloor w_j / \mu \rfloor U_j$, we have $\sum \lfloor w_j / \mu \rfloor U_j^0 \leq \sum \lfloor w_j / \mu \rfloor U_j^* \leq W^* / \mu$. Therefore, $W^0 \leq W^* + \epsilon L \leq (1 + \epsilon)W^*$. Thus, Algorithm $A_{\epsilon, \rho}(L, R)$ is a (ϵ, ρ) -approximation algorithm for the problem $1/(\sum v_j x_j \leq K, \sum w_j U_j)$.

By substituting R/μ for Y , K/δ for K , and K/δ for each τ_j in the time requirement of Algorithm $D1(Y)$, we obtain the time requirement of $A_{\epsilon, \rho}(L, R)$ as $O(n^2 R K^2 / (\mu \delta^2))$ which, on substitution of $\mu = \epsilon L/n$ and $\delta = \rho K/n$, yields the time bound as indicated in the theorem. $\square$

We note that a similar approach can be applied to modify $D2(X)$ to be a (ϵ, ρ) -approximation algorithm for the problem $1/(\sum v_j x_j, \sum w_j U_j \leq K)$.

Assume that $\min_{1 \leq j \leq n} \{w_j\} \leq W^*$. Otherwise, $W^* = 0$ is the optimal solution value and the problem reduces to one of minimizing $\sum v_j x_j$ subject to each job j meeting deadline d_j . As it has already been mentioned, the latter problem has been studied by Janiak and Kovalyov (1993). Since $W^* \leq \sum w_j$, we can set, without loss of generality, $L = \min_{1 \leq j \leq n} w_j$ and $R = \sum w_j$ in $A_{\epsilon, \rho}(L, R)$. We define a (ϵ, ρ) -approximation algorithm for the problem $1/(\sum v_j x_j \leq K, \sum w_j U_j)$ as Algorithm $A_{\epsilon, \rho}(L, R)$ with such values L and R . If $w_j = w$ for all j , then the time complexity of this algorithm is $O(n^6/(\epsilon \rho^2))$.

To construct the Pareto set ϵ -approximation P_ϵ for the problem $1/(\sum v_j x_j, \sum w_j U_j)$, we apply Algorithm B_ϵ presented in section 2. In this algorithm, the procedure $B(l)$ is our (ϵ, ρ) -approximation algorithm for the problem $1/(\sum v_j x_j \leq K, \sum w_j U_j)$ with $\rho = \epsilon/(2 + \epsilon)$. It should be noted that, in B_ϵ , a positive lower bound $L_1 > 0$ for the value of $\sum v_j x_j$ is assumed. Therefore, we first consider separately a Pareto optimal solution with zero resource allocation. This solution is an optimal solution to the problem of minimizing $\sum w_j U_j$ subject to the given processing times $p_j = b_j, j = 1, \dots, n$. For the latter problem, an approximation algorithm is presented by Gens and Levner (1981), which delivers a solution q with a value at most $(1 + \epsilon)$ times the optimal solution value in $O(n^2/\epsilon)$ time. Then we apply this algorithm, include q into P_ϵ , and set $L_1 = \min_{1 \leq j \leq n} \{v_j\} > 0$ in our general scheme B_ϵ .

It is easy to see that, with these modifications, the time complexity of the algorithm B_ϵ for the problem $1/(\sum v_j x_j, \sum U_j)$ is $O(n^6 \log(\sum v_j \tau_j)/\epsilon^3)$. In this time bound, ϵ is substituted for ρ following inequalities $\rho = \epsilon/(2 + \epsilon) \geq \epsilon/3$ for $\epsilon \leq 1$ and $\rho > 1/3$ for $\epsilon > 1$. Since B_ϵ is polynomial in the problem instance length and in $1/\epsilon$, the family of algorithms $\{B_\epsilon\}$ forms a *fully polynomial approximation scheme*.

REFERENCES

- [1] T. C. E. CHENG, Z.-L. CHEN, C. L. LI, AND B. M.T. LIN, *Single-machine scheduling to minimize the sum of compression and late costs*, Naval Res. Logist., 1998, to appear.
- [2] T. C. E. CHENG, Z.-L. CHEN, AND C. L. LI, *Single-machine scheduling with trade-off between number of tardy jobs and resource allocation*, Oper. Res. Lett., 19 (1996), pp. 237–242.
- [3] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability: A Guide of the Theory of NP-Completeness*, W. H. Freeman and Co., New York, 1979.
- [4] G. V. GENS AND E. V. LEVNER, *Fast approximation algorithm for job sequencing with deadlines*, Discrete Appl. Math., 3 (1981), pp. 313–318.
- [5] R. L. GRAHAM, E. L. LAWLER, J. K. LENSTRA, AND A. H. G. RINNOOY KAN, *Optimization and approximation in deterministic sequencing and scheduling: A survey*, Ann. of Discrete Math., 3 (1979), pp. 287–326.
- [6] P. HANSEN, *Bicriterion path problem*, in Lecture Notes in Econom. and Math. Systems 177, Springer-Verlag, New York, 1980, pp. 109–127.
- [7] J. A. HOOGEVEEN, *Single-Machine Bicriteria Scheduling*, Ph.D. thesis, CWI, Amsterdam, 1992.
- [8] A. JANIAK, *Single machine scheduling problem with a common deadline and resource dependent release dates*, European J. Operational Research, 53 (1991), pp. 317–325.
- [9] A. JANIAK AND M. Y. KOVALYOV, *Single Machine Scheduling with Deadlines and Resource Dependent Processing Times*, Working paper, Institute of Engineering Cybernetics, Technical University of Wroclaw, Wroclaw, Poland, 1993.
- [10] E. L. LAWLER, *Optimal sequencing of a single machine subject to precedence constraints*, Management Science, 19 (1973), pp. 544–546.
- [11] E. L. LAWLER, *A fully polynomial approximation scheme for the total tardiness problem*, Oper. Res. Lett., 1 (1982), pp. 207–208.
- [12] E. L. LAWLER AND J. M. MOORE, *A functional equation and its application to resource allocation and sequencing problems*, Management Science, 16 (1969), pp. 77–84.

- [13] C.-Y. LEE AND G. VAIRAKTARAKIS, *Single machine dual criteria scheduling: A survey*, in Complexity in Numerical Optimization, P. M. Pardalos, ed., World Scientific, River Edge, NJ, 1993, pp. 269–298.
- [14] J. M. MOORE, *An n job, one machine sequencing algorithm for minimizing the number of late jobs*, Management Science, 15 (1968), pp. 102–109.
- [15] E. NOWICKI AND S. ZDRZALKA, *A survey of results for sequencing problems with controllable processing times*, Discrete Appl. Math., 26 (1990), pp. 271–287.
- [16] M. H. ROTHKOPF, *Scheduling independent tasks on parallel processors*, Management Science, 12 (1966), pp. 437–447.
- [17] S. SAHNI, *General techniques for combinatorial approximation*, Oper. Res., 25 (1977), pp. 920–936.
- [18] W. E. SMITH, *Various optimizers for single-stage production*, Naval Res. Logist., 1 (1956), pp. 59–66.
- [19] R. G. VICKSON, *Choosing the job sequence and processing times to minimize total processing plus flow cost on a single machine*, Oper. Res., 28 (1980), pp. 1155–1167.
- [20] R. G. VICKSON, *Two single machine sequencing problems involving controllable job processing times*, AIIE Transactions, 12 (1980), pp. 258–262.
- [21] W. WILLBORN AND T. C. E. CHENG, *Global Management of Quality Assurance Systems*, McGraw-Hill, New York, 1994.
- [22] T. J. WILLIAMS, *Analysis and Design of Hierarchical Control Systems with Special Reference to Steel Plant Operations*, North-Holland, Amsterdam, 1985.

NORMAL-BOUNDARY INTERSECTION: A NEW METHOD FOR GENERATING THE PARETO SURFACE IN NONLINEAR MULTICRITERIA OPTIMIZATION PROBLEMS*

INDRANEEL DAS[†] AND J. E. DENNIS[‡]

Abstract. This paper proposes an alternate method for finding several Pareto optimal points for a general nonlinear multicriteria optimization problem. Such points collectively capture the trade-off among the various conflicting objectives. It is proved that this method is independent of the relative scales of the functions and is successful in producing an evenly distributed set of points in the Pareto set given an evenly distributed set of parameters, a property which the popular method of minimizing weighted combinations of objective functions lacks. Further, this method can handle more than two objectives while retaining the computational efficiency of continuation-type algorithms. This is an improvement over continuation techniques for tracing the trade-off curve since continuation strategies cannot easily be extended to handle more than two objectives.

Key words. Pareto set, multicriteria optimization, multiobjective optimization, trade-off curve

AMS subject classification. 90C29

PII. S1052623496307510

1. Introduction. A wide variety of problems arising in design optimization of engineering systems inherently involve optimizing multiple performance criteria (see, for example, Eschenauer, Koski, and Osyczka [3] and Statnikov and Matusov [4]). For example, a typical bridge-construction design might involve simultaneously minimizing the total mass of the structure and maximizing its stiffness. However, it is highly improbable that these conflicting objectives would both be “extremized” by the same design. Hence the designer makes some trade-offs among the conflicting objectives in choosing the final design.

In mathematical notation a multicriteria optimization problem can be loosely posed as

$$\text{“min”}_{x \in C} F(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_n(x) \end{bmatrix}, \quad n \geq 2, \quad \dots (MOP),$$

where

$$C = \{x : h(x) = 0, g(x) \leq 0, a \leq x \leq b\},$$

$F : \mathbb{R}^N \mapsto \mathbb{R}^n$, $h : \mathbb{R}^N \mapsto \mathbb{R}^{n_e}$, and $g : \mathbb{R}^N \mapsto \mathbb{R}^{n_i}$ are twice continuously differentiable mappings, and $a \in (\mathbb{R} \cup \{-\infty\})^N$, $b \in (\mathbb{R} \cup \{\infty\})^N$, N being the number of variables,

*Received by the editors July 22, 1996; accepted for publication (in revised form) April 1, 1997; published electronically June 3, 1998. This research was partially supported by Department of Energy, DOE grant DE-FG03-95ER25257, Air Force grant F49620-95-1-0210, and by the National Aeronautics and Space Administration contract NAS1-19480 while the first author was in residence at the Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, VA 23681-0001.

<http://www.siam.org/journals/siopt/8-3/30751.html>

[†]Department of Computational & Applied Mathematics, Rice University, Houston, TX 77251-1892. Current address: IBM, Hopewell Junction, NY 12533 (indra@caam.rice.edu).

[‡]Noah Harding Professor of Computational & Applied Mathematics, Rice University, Houston, TX 77251-1892 (dennis@rice.edu).

n the number of objectives, and ne and ni the number of equality and inequality constraints.

Since no single x^* would generally minimize every f_i simultaneously, a concept of optimality which is useful in the multiobjective framework is that of Pareto optimality, as explained below.

DEFINITION. *The vector $F(\hat{x})$ is said to dominate another vector $F(\bar{x})$, denoted $F(\hat{x}) \prec F(\bar{x})$, if and only if $f_i(\hat{x}) \leq f_i(\bar{x})$ for all $i \in \{1, 2, \dots, n\}$ and $f_j(\hat{x}) < f_j(\bar{x})$ for some $j \in \{1, 2, \dots, n\}$. A point $x^* \in C$ is said to be globally Pareto optimal or a globally efficient point for (MOP) if and only if there does not exist $x \in C$ satisfying $F(x) \prec F(x^*)$. $F(x^*)$ is then called globally nondominated or noninferior.*

Computational methods for general nonlinear multicriteria optimization, including the one described in this paper, can at best guarantee local Pareto optimality of the obtained solution. The definition of local Pareto optimality is very similar to its global counterpart: a point $x^* \in C$ is said to be locally Pareto optimal or a locally efficient point for (MOP) if and only if there exists an open neighborhood of x^* , $\mathcal{B}(x^*)$, such that there does not exist $x \in \mathcal{B}(x^*) \cap C$ satisfying $F(x) \prec F(x^*)$.

Pareto optimality will henceforth refer to local Pareto optimality unless qualified explicitly.

The shadow minimum or utopia point, F^* , is defined as the vector containing the individual global minima, f_i^* , of the objectives, i.e.,

$$F^* = \begin{bmatrix} f_1^* \\ f_2^* \\ \vdots \\ f_n^* \end{bmatrix}.$$

We assume here and henceforth the existence of a minimizer for each of our objectives. The shadow minimum could thus be attained only in the rare case when a single x minimizes all the objective functions. However, in practical situations, the best we can hope for is to get close to the shadow minimum and assure that there is an agreeable trade-off among the multiple objectives.

Very often in engineering applications the desired result helpful in facilitating design is a whole collection of Pareto optimal points, representative of the entire spectrum of efficient solutions. Thus, ideally, the desired solution is the entire Pareto optimal set, which can be obtained for some small problems that allow themselves to be treated parametrically, resulting in closed-form expressions for the Pareto set (see Lin [5]). More recently, attempts have been made to approximate the entire curve of Pareto optimal solutions in biobjective problems using techniques that trace the curve of parametrized optima (see Rakowska, Haftka, and Watson [6]; Rao and Papalambros [7]; Lundberg and Poore [8]).

Another alternative acceptable in most applications is a discrete set of Pareto optimal points obtained by combining the multiple objectives into a single objective function and minimizing the single objective over various values of the parameters used to combine the objectives. For example, it is possible to generate a set of Pareto optimal points by minimizing a convex combination of the objectives, $w^T F(x)$, over $x \in C$, where $w \geq 0$ (componentwise) and $\sum_{i=1}^n w_i = 1$, and by performing the minimization for different choices of w (see, among many others, Koski [14]). In this article, we propose a new method for generating Pareto optimal points which is at

least as efficient as these methods and, unlike the techniques for tracing the curve of Pareto optimal solutions, can be applied to problems with more than two objectives.

2. Preliminaries. First let us introduce some terminology.

Convex hull of individual minima (CHIM). Let x_i^* be the respective global minimizers of $f_i(x)$, $i = 1, \dots, n$ over $x \in C$. Let $F_i^* = F(x_i^*)$, $i = 1, \dots, n$. Let Φ be the $n \times n$ matrix whose i th column is $F_i^* - F^*$ sometimes known as the *pay-off matrix*. Then the set of points in $\mathbb{R}^n$ that are convex combinations of $F_i^* - F^*$, i.e., $\{\Phi\beta : \beta \in \mathbb{R}^n, \sum_{i=1}^n \beta_i = 1, \beta_i \geq 0\}$, is referred to as the CHIM.

The set of attainable objective vectors $\{F(x) : x \in C\}$ is denoted by $\mathcal{F}$, so $F : C \mapsto \mathcal{F}$; i.e., C is mapped by F onto $\mathcal{F}$. The space $\mathbb{R}^n$ which contains $\mathcal{F}$ is usually referred to as the *objective space*. The map of C under F in the objective space is often called the *multiloss map* (*biloss map*, if $n = 2$). We shall denote the boundary of $\mathcal{F}$ by $\partial\mathcal{F}$. The set of all Pareto optimal points is usually denoted by $\mathcal{P}$. The complete curve/surface of Pareto minima (continuous or not) is often referred to as the *trade-off function* (see p. 9, Haimes, Hall, and Freedman [15]).

CHIM₊: Let $CHIM_\infty$ be the affine subspace of lowest dimension that contains the CHIM, i.e., the set $\{\Phi\beta : \beta \in \mathbb{R}^n, \sum_{i=1}^n \beta_i = 1\}$. Then $CHIM_+$ is defined as the convex hull of the points in the intersection of $\mathcal{F}$ and $CHIM_\infty$. More informally, consider extending (or withdrawing) the boundary of the CHIM simplex to touch $\partial\mathcal{F}$; the “extension” of CHIM thus obtained is defined as $CHIM_+$.

Henceforth, it shall be assumed that the objective functions have been defined with the shadow minimum shifted to the origin, so that all the objective functions are nonnegative, i.e., $F(x)$ is redefined as

$$F(x) \leftarrow F(x) - F^*.$$

We observe that in Fig. 1, which shows the set $\mathcal{F}$ in the objective space, the point A is F_1^* , B is F_2^* , O is the shadow minimum (and the origin), the broken line segment AB is the CHIM, while the “arc” ACB is the set of all Pareto minima in the objective space, alternately, the trade-off curve. In this (and any) problem with $n = 2$ (i.e., biobjective), $CHIM = CHIM_+$. For $n > 2$ CHIM may not equal $CHIM_+$ as in the case shown in Fig. 3.

3. Central idea. Normal-boundary intersection (NBI) is a technique intended to find the portion of $\partial\mathcal{F}$ which contains the Pareto optimal points. In order to facilitate the introduction of the preliminary idea behind NBI the discussion will assume that the vector of global minima of the objectives, F^* , is available. In section 4.2 it will be argued how not having global minima usually renders very little injury to the technique.

The algebraic idea behind our approach will be motivated by means of a simple and obvious idea: the intersection point between the boundary $\partial\mathcal{F}$ and the normal pointing toward the origin emanating from any point in the CHIM is a point on the portion of $\partial\mathcal{F}$ containing the efficient points. This point is also a Pareto optimal point unless it happens to lie in a “sufficiently concave” part of the boundary as shown in Fig. 2. It certainly is a Pareto optimal point when the trade-off surface in the objective space is convex, which happens in almost every application found in the literature. If the trade-off surface is not convex, points in the concave part will still be obtained using NBI. If these points in the concave part are Pareto optimal this particular trait can be thought of as a merit of NBI over minimizing convex combinations of objectives which fails to obtain points in the nonconvex parts of the

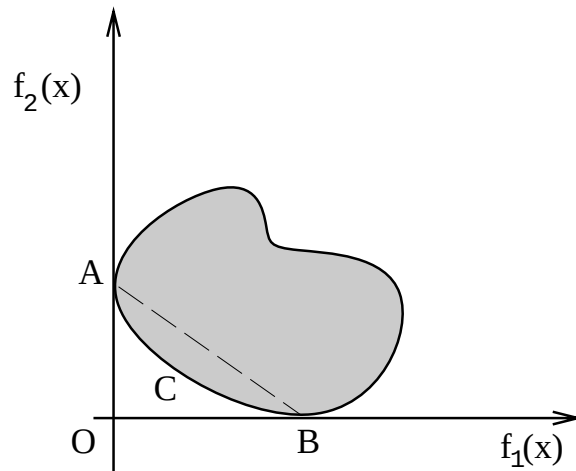


FIG. 1. A typical biloss map.

Pareto set (see Das and Dennis [2]). If they are not Pareto optimal this might be characterized as a disadvantage. Nevertheless these points are useful even though they are not Pareto optimal, since they help in constructing a smoother approximation of the Pareto boundary.

It should be noted that the *goal attainment method* described in Gembicki [9], or a very similar method in Schy and Giesy [10], [11], [13] and Schy, Giesy, and Johnson [12] can also be interpreted in terms of the geometrical idea described above.

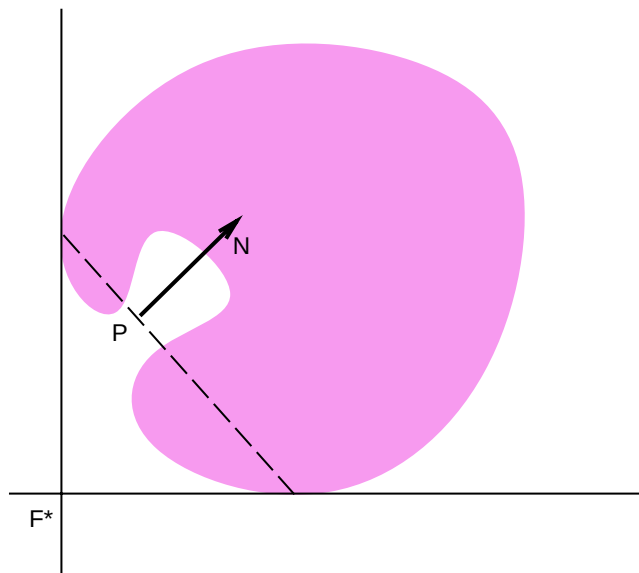


FIG. 2. Boundary point obtained by NBI is not Pareto optimal.

Now let us illustrate algebraically how any such boundary point can be found by solving an optimization problem. Given barycentric coordinates β , $\Phi\beta$ represents a

point in the *CHIM*. Let $\hat{n}$ denote the unit normal to the *CHIM* simplex pointing toward the origin; then $\Phi\beta + t\hat{n}, t \in \Re$ represents the set of points on that normal. The point of intersection of the normal and the boundary of $\mathcal{F}$ closest to the origin is the global solution of the following subproblem:

$$\begin{aligned}
 & \max_{x,t} t \\
 (1) \quad & \text{s.t.} \quad \Phi\beta + t\hat{n} = F(x), \\
 & h(x) = 0, \quad (NBI_\beta), \\
 & g(x) \leq 0, \\
 & a \leq x \leq b.
 \end{aligned}$$

The vector constraint $\Phi\beta + t\hat{n} = F(x)$ ensures that the point x is actually mapped by F to a point on the normal, while the remaining constraints ensure feasibility of x with respect to the original problem (*MOP*). Observe that if the origin is not shifted to F^* the first set of constraints should read $\Phi\beta + t\hat{n} = F(x) - F^*$.

The subproblem above shall be referred to as the *NBI subproblem* and be written as NBI_β since β is the characterizing parameter of the subproblem. Solutions of these subproblems will be referred to as *NBI points*. The idea is to solve NBI_β for various β and find several points on the boundary of $\mathcal{F}$, effectively constructing a pointwise approximation of the efficient frontier.

The goal attainment approach of Gembicki [9] or Schy and Giesy [10], [11], [13] results in a similar subproblem where the equality constraints (1) in the NBI subproblem get replaced by inequalities ($F(x) \leq u + tv$). However the work of Schy and Giesy was mainly concerned with finding one Pareto optimal point, so the concept of parametrizing the subproblem to generate many Pareto points was not studied. In their work, both the normal vector (v) and the point of origin of the normal (u) are user-defined quantities; once they are set, one Pareto point can be generated. On the other hand, NBI chooses a particular parametrization of the point of origin of the normal in terms of the barycentric coordinates β and keeps the normal direction $\hat{n}$ fixed. This particular parametrization plays a key role in generating the even spreads of Pareto points demonstrated later. Observe that unlike an NBI point, the solution of a goal attainment problem is not constrained to lie on the normal.

As indicated earlier, all NBI points are not Pareto optimal points. In biobjective problems, for every Pareto optimal point there exists a corresponding NBI subproblem of which it is the solution. The same is true for $n \geq 3$, with one difference: the coordinates of the parameter vector β for NBI_β may not be all nonnegative. As a simple example, suppose $\mathcal{F}$ is a sphere in $\Re^3$ touching the coordinate axes. Then the *CHIM* simplex is the triangle formed by joining the three points where the sphere touches the axes. Quite clearly $CHIM \neq CHIM_+$ so that there exist points in $CHIM_+ \setminus CHIM$ underneath which there are Pareto optimal points on the sphere. However, since these points are not in *CHIM*, they do not satisfy $\beta_i \geq 0 \forall i$. Thus, by solving NBI_β for $\sum_{i=1}^n \beta_i = 1, \beta_i \geq 0 \forall i$, a portion of the Pareto set might be overlooked for problems with $n > 2$. However, these overlooked points are likely to

be “extremal” Pareto points lying near the periphery of the Pareto surface and are not interesting from the trade-off standpoint, which is our primary goal. Figure 3 illustrates a similar situation. The reader interested in how these peripheral Pareto points can be obtained can look in Das [1] for such a technique interesting at least from a theoretical standpoint.

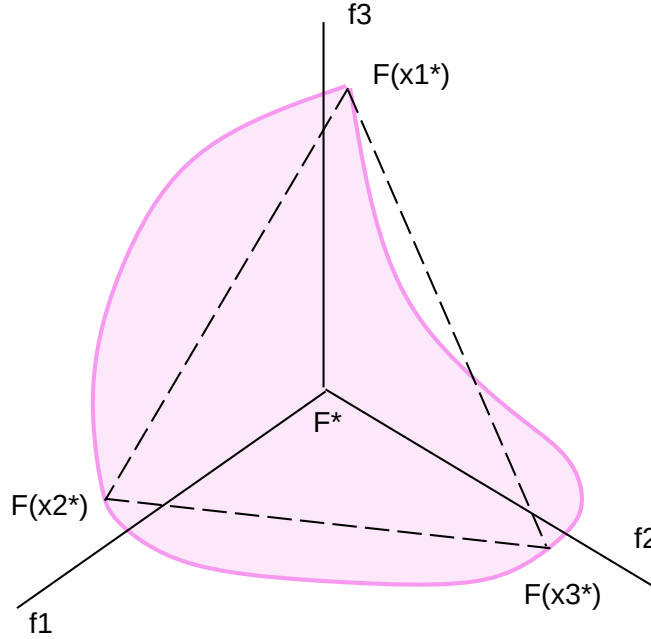


FIG. 3. There exist Pareto optimal points not obtainable using NBI.

4. Some details.

4.1. Structure of Φ . The i th column of Φ is described by

$$\Phi(:, i) = F(x_i^*) - F^*.$$

Since $f_i(x_i^*) = f_i^*$, clearly,

$$\Phi(i, i) = 0.$$

Moreover, since x_i^* is the minimizer of $f_i(x)$ over x_j^* , $j = 1, \dots, n$,

$$\Phi(j, i) \geq 0, j \neq i.$$

Thus a negative element in position (j, k) of Φ signifies that x_k^* is not the global minimizer of $f_k(x)$, and $f_k(x_j^*) < f_k(x_k^*)$; i.e., x_j^* improves on the current local minimum of $f_k(x)$. This fortunate occurrence provides a better starting point x_j^* for minimizing $f_k(x)$ and hence will lead to a better local minimum for f_k^* just by examining Φ .

4.2. Local versus global. As indicated earlier, most NBI points are guaranteed to be only locally Pareto optimal points. However, the components of the shadow

minimum F^* being global minima of the objectives and the Pareto surface being convex is a sufficient, although far from necessary, condition for the NBI points to be globally Pareto optimal. In situations like the one shown in Fig. 4, where the relevant part of $\partial\mathcal{F}$ is “folded,” the NBI point obtained may not be the one furthest out on the boundary along that normal because the solution of the nonlinear NBI subproblem is only guaranteed to be locally optimal. Thus the NBI point is not globally Pareto optimal.

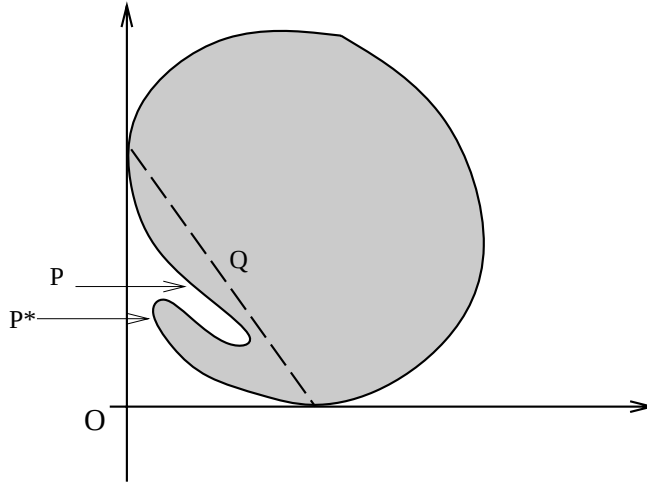


FIG. 4. NBI started at Q converges to P (locally Pareto optimal), whereas the corresponding globally efficient point would have been P^* .

Not being able to find globally Pareto optimal points is a drawback inherent in every method which finds a large number of efficient points of MOP. In homotopy methods, it would involve finding the global minimum of one of the two objectives in the very beginning. In methods which find efficient points by minimizing a single objective, only a global minimum of the scalarized objective would correspond to a globally efficient point.

Another important issue we had promised to deal with is the case when one or more components of the shadow minimum F^* consists of local but not global function minima. Such a case results in a different matrix Φ and more different goals $\Phi\beta$ for the NBI subproblems to improve on. These goals may be conservative or ambitious depending on the orientation of the incorrect $CHIM$ relative to the $CHIM$ formed using the true global minimizers. However, having the “incorrect $\Phi\beta$ ” may not preclude the NBI point from being a point on the efficient frontier, as in the case of Fig. 5. Once the globally efficient point P in Fig. 5 has been found, a trivial examination of its components reveals that the current x_1^* is not the global minimizer of f_1 and provides a starting point, viz., P , for restarting the nonlinear programming (NLP) to obtain a better local minimum of f_1 . Then NBI can be restarted with this improved estimate of F^* . Some (if not all) globally Pareto optimal points will be obtained in most problems even if NBI is not restarted. Some points that are not Pareto optimal may be obtained if the targets $\Phi\beta$ are conservative as in Fig. 5. In cases such as the one in Fig. 6, it is possible that all globally Pareto optimal points may not be found using NBI, and no indication regarding the local optimality of the function minima may be obtained.

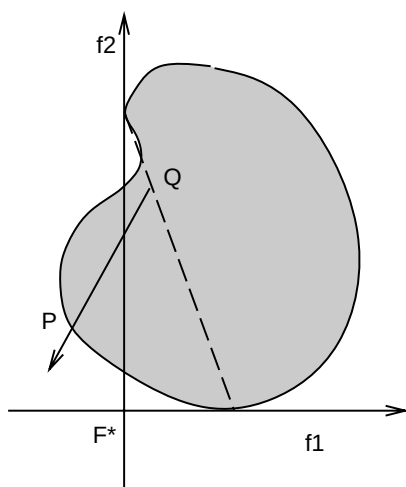


FIG. 5. NBI started at Q converges to globally Pareto optimal point P even though all the function minima in the components of F^* are not global minima.

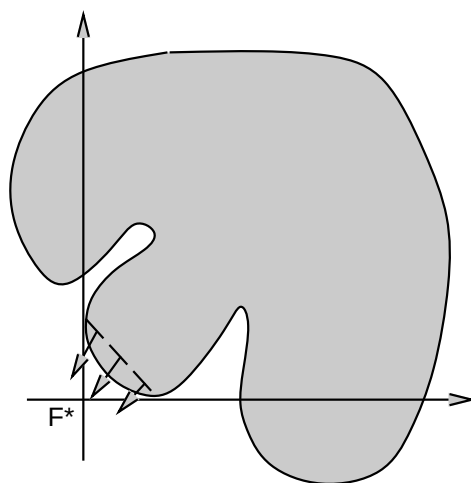


FIG. 6. Local minima in the components of F^* might prevent NBI from obtaining Pareto points from every part of the efficient frontier.

However, in situations like the ones in Fig. 7, owing to the fact that the individual function minima are only local, all the NBI points obtained are only locally Pareto optimal.

Computational experience (on more than just the problems mentioned here) shows that in cases where the global minima of the functions are not available at the onset, as NBI proceeds, either some component of Φ turns out to be negative or a function value of a particular objective is found that improves on its current local minimum value. This is not unusual given that the entire NBI procedure samples a large number of function values in the objective space.

To conclude this discussion and provide a general abstraction, it should be mentioned that whatever the components of F^* may be, NBI obtains at least the (local)

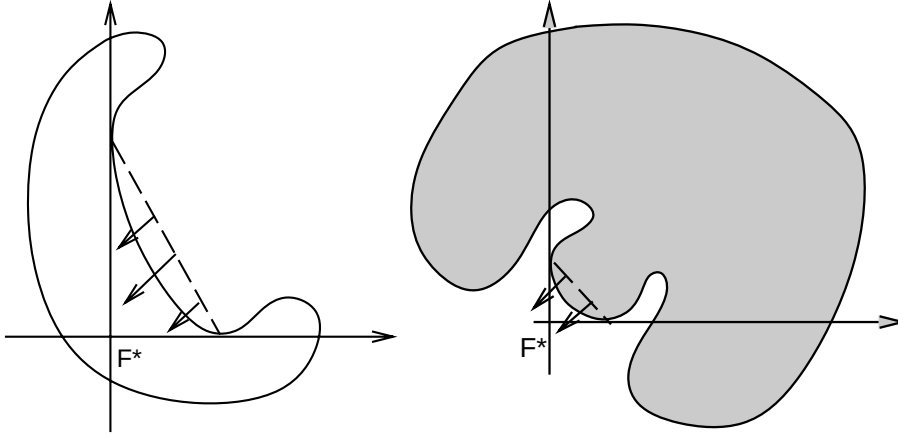


FIG. 7. In the first case $\mathcal{F} = \partial\mathcal{F}$. Here, having local function minima in the components of F^* can cause NBI to find only locally Pareto optimal points.

boundary points dominated by F^* unless F^* is attainable, i.e., $F^* \in \mathcal{F}$. If $F^* \in \mathcal{F}$, Φ has a column of zeros and/or NBI obtains some (local) boundary point which dominates F^* , providing reason to refine F^* and start NBI all over again.

4.3. Quasi-normal instead of normal direction. The idea of a family of normals intersecting the boundary is valid even if we do not have the exact normal direction to the *CHIM* simplex, but we have some *quasi-normal* direction $\hat{n}$ which has negative components; i.e., it points toward the origin. “Shooting” a family of quasi-normal rays toward the boundary also gets us our desired boundary points. In practice we choose our quasi-normal direction to be an equally weighted linear combination of the columns of Φ , multiplied by -1 to ensure that it points toward the origin. Explicitly,

$$\hat{n} = -\Phi e,$$

where e is the column vector of all ones.

The quasi-normal component defined as above has the property that the NBI point found for a certain β is completely independent of the scales of the objective functions. In other words, if NBI_β is re-solved with the objective functions rescaled by arbitrary factors, the NBI point found remains unchanged. This fact will be proved later.

Given that Φ has nonnegative components as discussed in the previous subsection, it is clear that all components of Φe are nonnegative.

Even though a quasi-normal direction will be used in our computations, we prefer to retain the name “NBI,” rather than change it to something like “QNBI” hoping this misnomer will not be judged too harshly.

4.4. Further insight: NBI and goal programming. Since t is being maximized in the NBI subproblem and $\Phi\beta + t\hat{n} = F(x)$, $x \in C$, this maximization subproblem attempts to find a feasible point x as far from a “target” point $\Phi\beta$ as possible, with $\hat{n} \leq 0$ (componentwise) guaranteeing nonincrease in the components of $F(x)$ relative to the components of $\Phi\beta$ if the optimal value of t is nonnegative.

This is similar to goal programming. If we take the Pareto surface to be convex

in the objective space, “equality goal programming”¹ can be thought of as NBI where the direction $\hat{n}$ is the negative of one of the canonical basis vectors e_i (i.e., with 1 in the i th position and 0 in the rest). To be precise, the subproblem NBI_β with $\hat{n} = -e_i$ has the same solution as the following goal programming problem:

$$\begin{aligned} & \min_x f_i(x) \\ \text{s.t.} \quad & f_j(x) = (\Phi\beta)(j), \quad j = 1, \dots, n, \quad j \neq i, \\ & x \in C, \end{aligned}$$

where $(\Phi\beta)(j)$ denotes the j th component of the vector $\Phi\beta$.

Although posing the goals as equalities is nontraditional, this equality constrained goal programming problem for obtaining a Pareto optimal point is discussed in Lin [5] and [16].

In a future section NBI will be related to the traditional goal programming problem using Lagrange multiplier theory without assuming that the Pareto surface is convex.

4.5. Efficiently solving the subproblems. The following simple observation plays a key role in lowering the computational expense involved in solving the NBI subproblems.

Consider parameter vectors β and $\bar{\beta}$ such that β is “close to” $\bar{\beta}$; i.e., $\|\beta - \bar{\beta}\|$ is “small” in some norm. Then it is reasonable to expect that the solution (x^*, t^*) of NBI_β and the solution $(\bar{x}^*, \bar{t}^*)$ of $NBI_{\bar{\beta}}$ are “close to each other.” Assume that we have solved $NBI_{\bar{\beta}}$ first and already have the point $(\bar{x}^*, \bar{t}^*)$. Then with $(\bar{x}^*, \bar{t}^*)$ as the starting point for solving NBI_β , the NBI subproblem solver can be expected to converge in relatively few iterations. It is this aspect of our algorithm that gives it the flavor of a continuation-type method.

Since we already have the individual minima of the functions, i.e., the vertices of the *CHIM* simplex, we start at x_1^* and solve a “nearby subproblem,” and then a subproblem close to the one just solved, and so on.

Of course “ordering the subproblems” may not be obvious for problems with more than two objective functions, but it can still be achieved, as described in the next section.

5. Generating β and ordering the subproblems for more than two objectives. In this section we shall describe a (data) structure which simultaneously enables the generation of weights β and the ordering of the subproblems in a manner amenable not only to efficient solution but also to parallelization.

5.1. Generating β . Let us assume that for an n -objective problem, $\delta_j > 0$ is the uniform spacing between two consecutive β_j values (i.e., the “stepsize” on the j th component of β) for $j = 1, \dots, n-1$. For simplicity, let us also assume that $\frac{1}{\delta_1}$ is an integer.

The possible values that can be assumed by β_1 are

$$[0, \delta_1, 2\delta_1, \dots, 1].$$

¹Referring to goal programming where the goal constraints are equalities instead of inequalities.

Given a particular value of β_1 , define $m_1 = \frac{\beta_1}{\delta_1}$. Then the possible values of β_2 corresponding to that value of β_1 (i.e., $\beta_1 = m_1\delta_1$; all the β_i 's must add up to 1) are

$$[0, \delta_2, 2\delta_2, \dots, k_2\delta_2],$$

where $k_2 = I[\frac{1-\beta_1}{\delta_2}] = I[\frac{1-m_1\delta_1}{\delta_2}]$.

Now define $m_2 = \frac{\beta_2}{\delta_2}$. Then the possible values of β_3 corresponding to $\beta_1 = m_1\delta_1$ and $\beta_2 = m_2\delta_2$ are

$$[0, \delta_3, 2\delta_3, \dots, k_3\delta_3],$$

where $k_3 = I[\frac{1-\beta_1-\beta_2}{\delta_3}] = I[\frac{1-m_1\delta_1-m_2\delta_2}{\delta_3}]$.

Thus, corresponding to $\beta_i = m_i\delta_i$, $i = 1, \dots, j-1$, the possible values of β_j for $j = 2, \dots, n-1$ are

$$[0, \delta_j, 2\delta_j, \dots, k_j\delta_j],$$

where

$$k_j = I\left[\frac{1 - \sum_{i=1}^{j-1} m_i\delta_i}{\delta_j}\right].$$

Finally the last component of β is defined as

$$\beta_n = 1 - \sum_{i=1}^{n-1} \beta_i.$$

The entire data structure above can be thought of as a tree where the number of children varies with the node and generation. Each generation or level represents a component of β , and each path from the root to the leaf represents a possible β vector. However, a tree structure is unnecessary for implementation; all that require storage are the numbers δ_j . Nevertheless the tree is useful as a conceptual aid.

Of the subproblems generated by the weights in the above tree, n subproblems (with $\beta = e_i$) have already been solved in the course of finding F^* . It should also be noted that since $\frac{\delta_i}{\delta_j}$ is not necessarily an integer $\forall i < j$, the spacings between the "last two" values of β_n may not be uniform.

5.2. Special case. For equal stepsizes on all β_i ,

$$\text{let } \delta_i = \delta, i = 1, \dots, n-1.$$

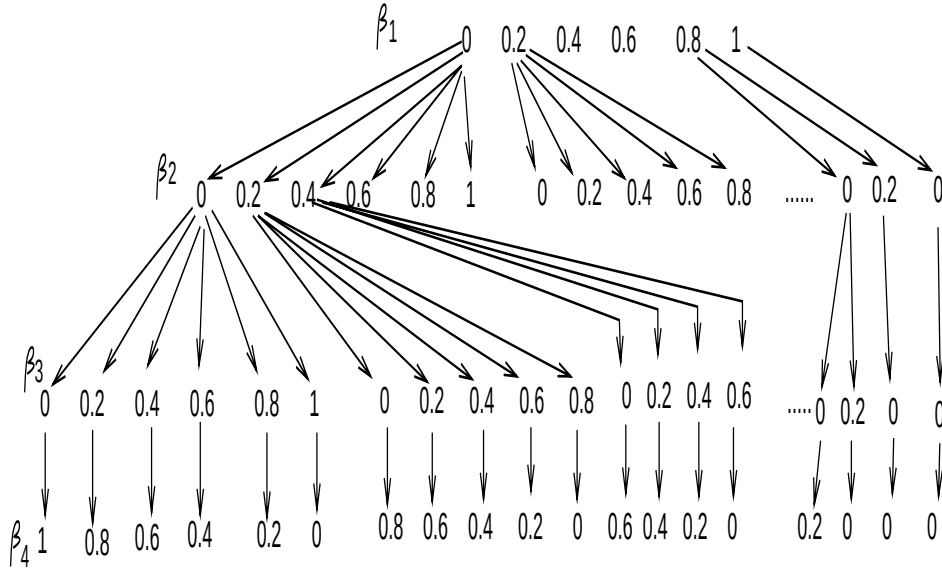
Also assume that $\frac{1}{\delta} = p$ is an integer.

As before, the possible values of β_1 are

$$[0, \delta, 2\delta, \dots, 1].$$

Then the possible values of β_j corresponding to $\beta_i = m_i\delta_i$, $i = 1, \dots, j-1$ for $j = 2, \dots, n-1$ are

$$\left[0, \delta, 2\delta, \dots, \left(p - \sum_{i=1}^{j-1} m_i\right)\delta\right].$$

FIG. 8. Generating β for $n = 4$, $\delta = 0.2$.

As before, $\beta_n = 1 - \sum_{i=1}^{n-1} \beta_i$, and now all the β_n values are uniformly spaced.

Figure 8 shows part of the tree of β values for $n = 4$ and $\delta = 0.2$.

Number of NBI subproblems. In general, the number of NBI subproblems for a given n and a given $p = \frac{1}{\delta}$ is given by

$$\binom{n+p-1}{p}.$$

In spite of the fact that we only intend to solve “nearby subproblems,” the computational cost of solving a huge number of NLP problems can be quite daunting. This motivates the need for parallelization, as will be mentioned in the next section.

5.3. Ordering the subproblems. Each path from the root of the tree (the topmost node) to a leaf (a member of the bottommost generation) represents a unique weight β . It should also be observed that the β vectors are already ordered on the basis of “nearness” as one traverses the tree breadthwise. Thus a strategy for picking the order of the subproblems could be to start with the leftmost one (which has $\beta = e_n$ and is already solved) and solve the next one in the β_{n-1} generation (which is $\beta_{n-1} = \delta_{n-1}, \beta_n = 1 - \delta_{n-1}$), then the next one in the β_{n-1} generation ($\beta_{n-1} = 2\delta_{n-1}, \beta_n = 1 - 2\delta_{n-1}$), and so on until all the subproblems for $\beta_i = 0, i = 1, \dots, n-2$ have been solved. Then we move to the next node in the β_{n-2} generation (i.e., with $\beta_i = 0, i = 1, \dots, n-3, \beta_{n-2} = \delta_{n-2}$) and visit all the children of this node, with the starting points of the NBI subproblems chosen as the corresponding NBI subproblem solutions at the previous node.

This is where the scope for parallelization comes in. The solution of the first subproblem at the second node in the β_{n-2} generation did not have to wait until all the subproblems in the first node were solved. The first subproblem in the second node of the β_{n-2} generation with $\beta_{n-2} = \delta_{n-2}, \beta_{n-1} = \delta_{n-1}, \beta_n = 1 - \delta_{n-2} - \delta_{n-1}$ can be solved immediately after solving the first subproblem in the first node with

$\beta_{n-2} = 0$, $\beta_{n-1} = \delta_{n-1}$, $\beta_n = 1 - \delta_{n-1}$. Thus the first subproblem in the second node can be solved in parallel with the second subproblem in the first node, $\dots$, and the k th subproblem in the second node can be solved in parallel with the $(k+1)$ th subproblem of the first node. Further, the k th subproblem in the third node can be solved in parallel with the $(k+1)$ th subproblem of the second node, with the solution of the k th subproblem of the second node as the starting point, and so on. This entire process of efficient parallelization is one of the topics of future research.

6. Relationship between the NBI subproblem and minimizing a convex combination of the objectives. In this section we illustrate how the NBI subproblem is related to the popular method of minimizing a convex combination of the objectives. This demonstrates how to go back and forth between the NBI parameter β and the convex combinations weight vector w for a particular Pareto point. The following discussion also demonstrates that corresponding to every w there exists a β such that NBI_β has the same solution as LC_w , but the converse is not true. In other words, this proves that there might be points obtainable using NBI not obtainable by minimizing convex combinations.

Given a Pareto point x^* , the problem can be thought of as being constrained only by the vector of equalities and binding inequalities and bounds at x^* . Let us denote this augmented vector of equalities by $\bar{h}(x)$. Let $w \in (\mathbb{R}_+ \cup \{0\})^n$, $\sum_1^n w_i = 1$, denote a positive, convex weighting of the objectives. The weighted linear combination problem for obtaining a Pareto optimal point is then written as

$$(2) \quad \begin{aligned} \min_x \quad & w^T F(x) \\ \text{s.t.} \quad & \bar{h}(x) = 0. \end{aligned}$$

The solution of the problem above will be referred to as an *LC point*, and the problem will be denoted by LC_w . Part of the first-order necessary or KKT condition for optimality of (x^*, λ^*) for problem (2) is

$$(3) \quad \nabla_x F(x^*)w + \nabla_x \bar{h}(x^*)\lambda^* = 0.$$

Similarly, if β denotes the vector of parameters in NBI_β , the NBI subproblem can be written as

$$(4) \quad \begin{aligned} \min_{x,t} \quad & -t \\ \text{s.t.} \quad & F(x) - \Phi\beta - t\hat{n} = 0, \\ & \bar{h}(x) = 0. \end{aligned}$$

Part of the KKT condition for optimality of $(x^*, t^*, \lambda^{(1)*}, \lambda^{(2)*})$ is

$$(5) \quad \nabla_x F(x^*)\lambda^{(1)*} + \nabla_x \bar{h}(x^*)\lambda^{(2)*} = 0,$$

$$-1 + \hat{n}^T \lambda^{(1)*} = 0,$$

where $\lambda^{(1)} \in \mathbb{R}^n$ represents the vector of multipliers corresponding to the constraints $\Phi\beta + t\hat{n} - F(x) = 0$, and $\lambda^{(2)} \in \mathbb{R}^{ne}$ denotes the multipliers of the equality constraints $\bar{h}(x) = 0$.

Claim. Suppose $(x^*, t^*, \lambda^{(1)*}, \lambda^{(2)*})$ is the solution of NBI_β and $\sum_1^n \lambda_i^{(1)*} \neq 0$. Now define the components of the vector w as

$$w_i = \frac{\lambda_i^{(1)*}}{\sum_1^n \lambda_i^{(1)*}}.$$

Then problem (2) with the above convex weighting vector w has the solution

$$\left[x^*, \lambda^* = \frac{1}{\sum_1^n \lambda_i^{(1)*}} \lambda^{(2)*} \right].$$

Proof. Dividing both sides of (5) by the scalar $\sum_1^n \lambda_i^{(1)*}$ and observing that $\bar{h}(x^*) = 0$, the equivalence between (3) and (5) becomes obvious.

However, quite clearly, if, for some i , the sign of $\lambda_i^{(1)*}$ is opposite to that of $\sum_1^n \lambda_i^{(1)*}$, then the vector w has a negative component and does not qualify as a weight for problem (2). In such a case, either the Pareto optimality of the NBI point $(x^*, t^*, \lambda^{(1)*}, \lambda^{(2)*})$ is questionable or the Pareto point lies in a nonconvex part of the Pareto set (Pareto points in nonconvex parts of the Pareto set cannot be obtained by minimizing a linear combination of the objectives).

Just as the above analysis gives a method for obtaining w for problem LC_w given the corresponding solution of NBI_β , one can also obtain the NBI point corresponding to a given solution of problem LC_w with very little effort.

Claim. Suppose (x^*, λ^*) solves problem LC_w . Let $(\bar{\beta}, t^*)$ be the solution of the $(n+1) \times (n+1)$ linear system

$$\Phi\beta + t\hat{n} = F(x^*),$$

$$\sum_{i=1}^n \beta_i = 1.$$

Then (x^*, λ^*) corresponds to the solution of NBI_β with $\beta = \bar{\beta}$; i.e., the solution of $NBI_{\bar{\beta}}$ is

$$\left(x^*, t^*, \lambda^{(1)*} = \frac{w}{w^T \hat{n}}, \lambda^{(2)*} = \frac{\lambda^*}{w^T \hat{n}} \right).$$

Proof. Let us divide (3) on both sides by $w^T \hat{n}$. This can always be done because w has nonnegative components (not all zero) and $\hat{n}$ has negative components, $w^T \hat{n} < 0$. Observing that $\lambda^{(1)*}$ defined above satisfies $\hat{n}^T \lambda^{(1)*} = 1$, we can see that the first part of the KKT conditions for $NBI_{\bar{\beta}}$ holds. Further observing that $\bar{h}(x^*) = 0$ and $\Phi\beta + t\hat{n} = F(x^*)$, the required equivalence between LC_w and $NBI_{\bar{\beta}}$ follows.

7. Relationship between the NBI subproblem and goal programming using multipliers. A solution to an NBI subproblem is also a solution to a goal programming problem given that some assumptions hold. This is elaborated on below, using the same type of multiplier argument used to relate NBI_β to LC_w .

Claim. Suppose $(x^*, t^*, \lambda^{(1)*}, \lambda^{(2)*})$ is the solution of NBI_β . Suppose that the components of $\lambda^{(1)*}$ are all of the same sign with at least one nonzero component. If $\lambda_k^{(1)*}$ is any such nonzero component, then x^* solves the following goal programming problem:

$$\begin{aligned} & \min f_k(x) \\ \text{s.t.} \quad & f_i(x) \leq \gamma_i, \forall i \neq k, \\ (6) \quad & \bar{h}(x) = 0 \end{aligned}$$

with goals γ_i given by

$$\gamma_i = \begin{cases} f_i(x^*) & \text{if } \lambda_i^{(1)*} \neq 0, \\ \text{any finite number } \geq f_i(x^*) & \text{if } \lambda_i^{(1)*} = 0 \end{cases}$$

$\forall i \in \{1, 2, \dots, n\} \setminus \{k\}$.

Proof. Since $(x^*, t^*, \lambda^{(1)*}, \lambda^{(2)*})$ solves the NBI_β subproblem, it must satisfy (5). Given that $\lambda_k^{(1)*} \neq 0$, we can divide both sides of (5) and get

$$\nabla_x f_k(x^*) + \sum_{i=1, i \neq k}^n \nabla_x f_i(x^*) \frac{\lambda_i^{(1)*}}{\lambda_k^{(1)*}} + \nabla_x \bar{h}(x) \frac{\lambda^{(2)*}}{\lambda_k^{(1)*}} = 0.$$

Now $\frac{\lambda_i^{(1)*}}{\lambda_k^{(1)*}} \geq 0$ because $\lambda_i^{(1)*}$ and $\lambda_k^{(1)*}$ are of the same sign. Then with $\frac{\lambda_i^{(1)*}}{\lambda_k^{(1)*}}$ as the multipliers of the $n - 1$ inequality constraints in (6), the goals γ_i satisfy complementarity by definition, since

$$\begin{aligned} \gamma_i &= f_i(x^*) \text{ whenever } \lambda_i^{(1)*} \neq 0 \\ \Rightarrow \frac{\lambda_i^{(1)*}}{\lambda_k^{(1)*}} (f_i(x^*) - \gamma_i) &= 0 \quad \forall i \neq k. \end{aligned}$$

Moreover, since x^* is clearly feasible for (6), $(x^*, \frac{\lambda_i^{(1)*}}{\lambda_k^{(1)*}}, \frac{\lambda^{(2)*}}{\lambda_k^{(1)*}})$ solves (satisfies first-order necessary conditions for minimizer for) problem (6).

8. Proof of independence with respect to function scales using the quasi-normal. In this section we shall prove that the NBI point found using the quasi-normal $\hat{n}$ and a particular β is independent of how the individual functions are scaled.

Let the objective functions be scaled by positive scalars s_i as

$$f_i(x) \leftarrow s_i f_i(x), \quad i = 1, \dots, n.$$

In other words, if s is the vector with components s_i and $S = \text{diag}(s)$, then

$$F(x) \leftarrow SF(x).$$

Consequently

$$\nabla_x F(x) \leftarrow \nabla_x F(x)S,$$

$$\Phi = S\Phi.$$

The quasi-normal direction $\hat{n} = -\Phi e$ after scaling becomes $= -S\Phi e$.

Claim. If $(x^*, t^*, \lambda^{(1)*}, \lambda^{(2)*})$ solves the unscaled NBI_β (i.e., with $S = I_n$), then $(x^*, t^*, S^{-1}\lambda^{(1)*}, \lambda^{(2)*})$ solves² NBI_β with the i th function scaled by s_i as above.

Proof. Since $(x^*, t^*, \lambda^{(1)*}, \lambda^{(2)*})$ solves the unscaled NBI_β (still with only equality constraints as in the previous section),

$$\nabla_x F(x^*)\lambda^{(1)*} + \nabla_x \bar{h}(x^*)\lambda^{(2)*} = 0,$$

$$\hat{n}^T \lambda^{(1)*} = 1,$$

$$\Phi\beta + t^*\hat{n} = F(x^*),$$

$$\bar{h}(x^*) = 0.$$

The first equation can be rewritten to state that the following holds:

$$(7) \quad (\nabla_x F(x^*)S)(S^{-1}\lambda^{(1)*}) + \nabla_x \bar{h}(x^*)\lambda^{(2)*} = 0.$$

The second equation implies

$$e^T \Phi^T \lambda^{(1)*} = 1$$

$$\equiv e^T \Phi^T S S^{-1} \lambda^{(1)*} = 1.$$

Since $S = S^T$, the above is the same as

$$(8) \quad (e^T (S\Phi)^T)(S^{-1}\lambda^{(1)*}) = 1.$$

The third equation can be rewritten as

$$\Phi\beta + t^*\Phi e = F(x^*)$$

$$(9) \quad \equiv S\Phi\beta + t^*S\Phi e = SF(x^*).$$

Clearly, equations (7), (8), and (9) imply that $(x^*, t^*, S^{-1}\lambda^{(1)*}, \lambda^{(2)*})$ solves NBI_β with the functions scaled by S . $\square$

The above result does not depend on e being the vector of all ones and consequently holds if $\hat{n}$ is scaled by a factor, say, a normalization constant.

The above result suggests that no matter how disparately the different functions might be scaled, NBI with the quasi-normal finds a set of points as if the functions were all scaled to the same order of magnitude.

²Here “solves” means “finds a stationary point of the NLP problem.”

9. Advantages of NBI.

• **Finds a uniform spread of Pareto points.** Consider any method that attempts to capture the shape of the Pareto surface by generating many points on the surface. An important property that would make such a method desirable is that it should generate an even spread of Pareto points, representative of all parts of the Pareto set, and not clusters of points in certain parts which fail to provide a good idea of the entire shape. Given that we can only solve a limited number of NLP subproblems and hence generate only a limited number of Pareto points, it becomes crucial to have the points spread as evenly as possible, so that a good approximation of the Pareto surface is obtained by solving as few subproblems as possible.

In implementing NBI, various settings of the parameter β are chosen such that the points $\Phi\beta$ form a uniformly spaced grid on the *CHIM* simplex (this is achieved by generating β as in section 5.2). Since the NBI points are restricted to lie on a set of parallel normals emanating from these “uniformly spread” points, the projections of the areas between neighboring NBI points on the *CHIM* are uniformly spread. Thus NBI can yield a good approximation of the Pareto surface by solving fewer NLP problems than weighted convex combinations. It is very difficult to guess the parameter settings for which weighted convex combinations yield a uniform spread of Pareto points because the weights that correspond to an even spread depend on the shape of the Pareto surface, as shown in Das and Dennis [2].

The interrelationship between the linear combinations subproblem and the NBI subproblem provides more insight into why the linear combinations technique fails to give a uniformly distributed set of Pareto optima. By fixing the weights w in subproblem LC_w , in effect the multipliers of the corresponding NBI subproblem get fixed, thus partly restricting the solution of the resultant subproblem. Even if the Pareto optima are uniformly distributed in the Pareto set, there is no reason why the corresponding multipliers should be uniformly distributed. More insight into the failures of convex combinations can be found in Das and Dennis [2].

However, the weights in the linear combinations approach are often very desirable because they give an idea of the relative importance of the objectives. Thus obtaining the NBI points, which are uniformly distributed, and then finding the corresponding weights w for the NBI points can be quite insightful.

• **Improves over homotopy techniques.** NBI improves over homotopy/continuation techniques for tracing the curve of Pareto optimal solutions like the one discussed in Rakowska, Haftka, and Watson [6] in the following respects.

- *Applicable for more than two objectives.* NBI is formulated to handle an arbitrary number of objectives. On the other hand, for a multiobjective problem with more than two objectives the homotopy parameter is not a scalar and the associated differential equation is a system of nonlinear partial differential equations with not readily available boundary conditions, rather than an ordinary initial value problem, as in the case of two objectives.
- *Does not require exact Hessian.* Even for a biobjective problem, solving the homotopy initial value problem requires exact second derivative information (i.e., the Hessian of the Lagrangian), whereas the NBI subproblem solver can use any NLP technique. Even if the NLP technique for the NBI subproblem requires gradient information, secant methods for NLPs make exact Hessians unnecessary.
- *Bypasses tracking active sets.* For problems with inequality constraints or explicit bounds on variables, homotopy techniques need to keep track of the

changes in active sets of the inequality constraints or bounds meticulously in the course of the numerical integration, which can present difficulties if the number of inequalities or bounds is large. On the other hand, an interior point NLP solver used as the NBI subproblem solver would handle this situation quite efficiently and would not have a problem with frequent changes in the active set.

It must be noted though that points where the active set changes provide important information to the designer. However, homotopy needs to keep track of changes in active sets even in the uninteresting parts of the Pareto set, whereas once the NBI points are found it is not difficult to trace how the active set changes along the Pareto surface by examining the binding inequalities at the Pareto points.

- *Does not assume connectedness or smoothness of the Pareto set.* The homotopy technique assumes that the Pareto curve is continuous and differentiable and is also connected to be able to integrate along the curve. This is not the case with NBI, although it might end up reporting some subproblems as infeasible if the Pareto set is disconnected.

- NBI improves on other traditional methods like goal programming in the sense that it never requires any prior knowledge of “feasible goals.” It improves on multilevel optimization techniques from the trade-off standpoint, since multilevel techniques usually can improve only a few of the “most important” objectives, leaving no compromise for the rest.

10. A numerical example. Below is a brief account of employing NBI techniques on a small biobjective problem stated below:

$$\min_x \begin{bmatrix} f_1(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 \\ f_2(x) = 3x_1 + 2x_2 - \frac{x_3}{3} + 0.01(x_4 - x_5)^3 \end{bmatrix}$$

$$\text{s.t.} \quad x_1 + 2x_2 - x_3 - 0.5x_4 + x_5 = 2,$$

$$4x_1 - 2x_2 + 0.8x_3 + 0.6x_4 + 0.5x_5^2 = 0,$$

$$x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 \leq 10.$$

NBI using the quasi-normal was run on the above problem with the evenly spread values of β with $\delta = 0.05$. The Pareto points thus obtained are tabulated below in Table 1 and are plotted in Fig. 9.

The method of convex combinations was run thrice on the same problem, with the weight vectors w assuming the same 21 uniformly spread values as the w vector above. The efficient solution scheme, i.e., starting the solution of a subproblem from the optimal point of the “nearest subproblem,” was used here as well.

When run on the original problem the minimizer of $f_2(x)$ was found six times for six different w , and there was a considerable gap “in the middle” of the Pareto set (see Fig. 9).

With f_1 scaled by 5, the point found six times earlier was found only twice (i.e., heavily weighting the first objective made the minimizer move away from x_2^*), but the Pareto optimal vectors obtained were concentrated at the $F(x_1^*)$ end and no “middle ground for compromise” was captured.

TABLE 1

β	Objective values
0.00, 1.00	10.0000, -4.0111
0.05, 0.95	9.4254, -3.7706
0.10, 0.90	8.8546, -3.5276
0.15, 0.85	8.2882, -3.2818
0.20, 0.80	7.7264, -3.0329
0.25, 0.75	7.1698, -2.7807
0.30, 0.70	6.6189, -2.5247
0.35, 0.65	6.0743, -2.2647
0.40, 0.60	5.5368, -2.0000
0.45, 0.55	5.0072, -1.7302
0.50, 0.50	4.4866, -1.4546
0.55, 0.45	3.9764, -1.1722
0.60, 0.40	3.4781, -0.8820
0.65, 0.35	2.9939, -0.5827
0.70, 0.30	2.5266, -0.2724
0.75, 0.25	2.0801, 0.0514
0.80, 0.20	1.6597, 0.3922
0.85, 0.15	1.2740, 0.7556
0.90, 0.10	0.9370, 1.1506
0.95, 0.05	0.6754, 1.5947
1.00, 0.00	0.5551, 2.1306

With f_1 scaled by 10, the point repeated earlier was found only once, although the clustering at the $F(x_1^*)$ end increased (see Fig. 10).

The Pareto optimal vectors obtained using linear combinations are shown in Table 2.

Clearly the inability of the method of convex combinations in adequately capturing the shape of the Pareto surface renders it fairly useless as a means of studying the trade-off between the conflicting objectives.

11. A truss optimization problem. Now we shall apply NBI to a truss optimization problem, a version of which has been studied in Koski [14]. The problem involves optimizing the design of a pin-jointed linear truss structure as shown in Fig. 11.

The problem is to find the optimal position of the vertical bar of fixed length L (the bars on the edge get fixed and their lengths decided accordingly) between $1/4$ and $3/4$ of the entire distance D and the optimal bar cross-sectional areas. The angles θ and α clearly depend on the chosen location x . Other optimization variables are the cross-sectional areas of the bars, a_1, a_2, a_3 , allowed to vary between 0.8 in^2 and 3.0 in^2 .

The objectives to be considered for minimization are the total volume of the structure, the displacement of the node, and the absolute value of the stress in each of the three bars. In the structure considered by Koski in [14] the location of the middle bar was fixed, so that θ and α were also fixed. Also, his total volume was a linear function of the design variables, unlike in our formulation where total volume is expressed as $a_1 \frac{L}{\sin \theta} + a_2 L + a_3 \frac{L}{\sin \alpha}$.

Without going into further details of the problem and the data involved, which can be found in Das [1], we present some Pareto plots for subsets of the five objectives mentioned here. Figure 12 shows the Pareto curve for minimizing the square of nodal displacement and the total volume with constraints on the absolute stresses in the three bars. The apparently unexpected gaps in the Pareto curve using NBI are points corresponding to which the NBI subproblems were infeasible owing to discontinuities

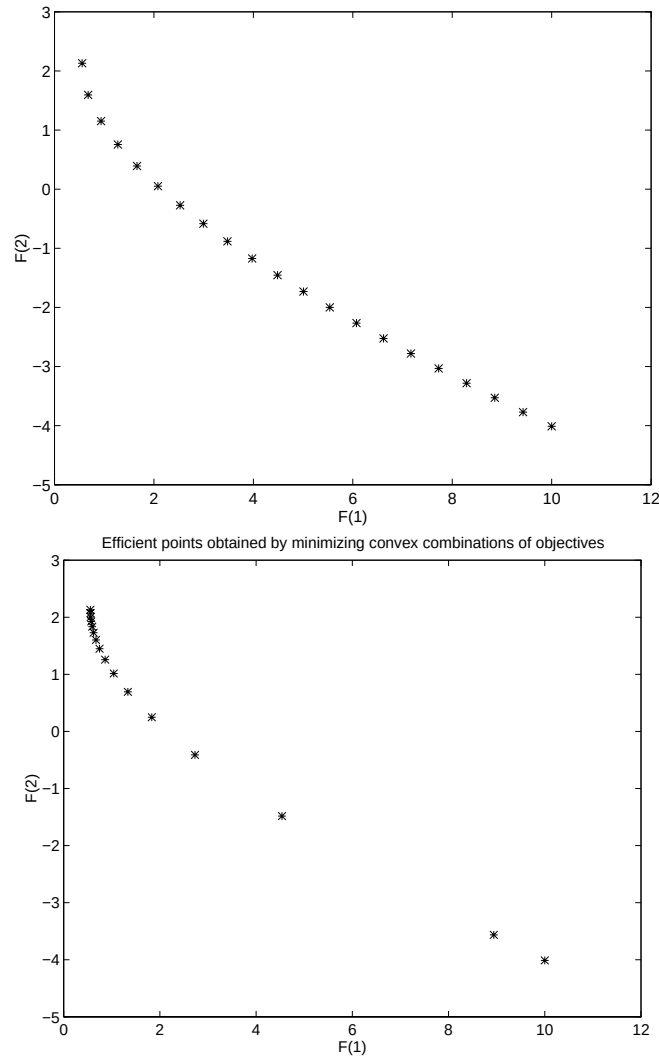


FIG. 9. *Pareto optimal vectors in the objective space using NBI and the method of convex combinations, respectively.*

in the Pareto set introduced by stringent stress inequalities.

Figure 13 shows the Pareto curve for minimizing the stress in the right bar (the minimum value of the stress was positive and hence the absolute sign was dropped) and the total volume with constraints on the absolute stresses in the middle and left bars.

Given the individual minima and minimizers of the objectives at the outset, the number of floating point operations (flops) required in solving the subproblems for minimizing the stress in the right bar and the total volume using NBI and convex combinations for 21 parameter settings for each are shown in Table 3.

Table 3 shows that NBI takes about twice as many flops but finds about twice as many distinct points, so that the number of flops per Pareto point is almost the same for the two methods (convex combinations wins marginally). But NBI yields a

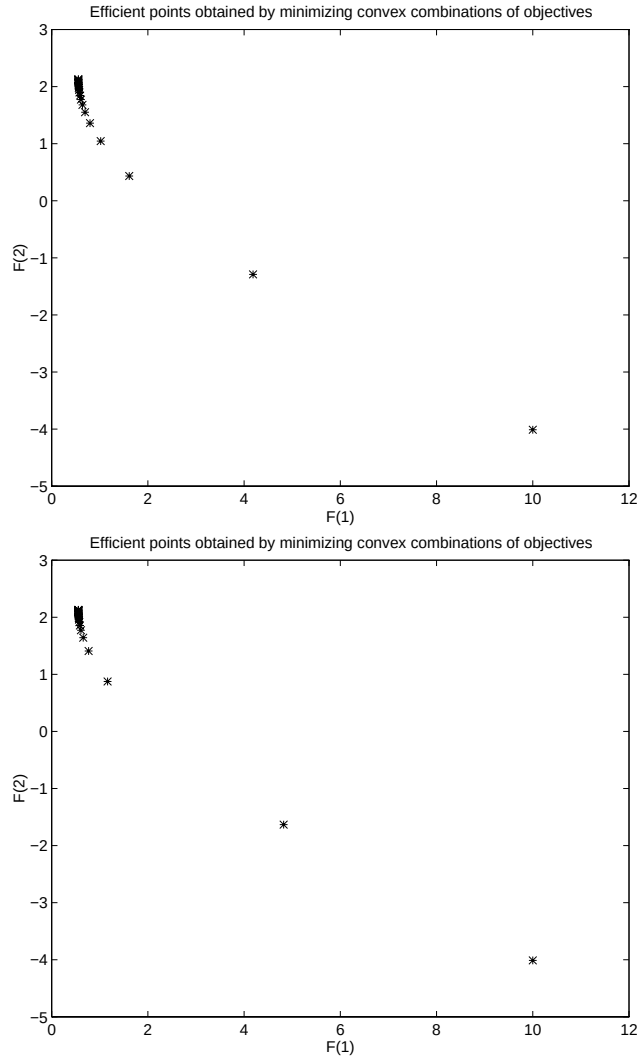


FIG. 10. Pareto optimal vectors in the objective space using the method of linear combinations on the problem with f_1 scaled by 5 and 10, respectively.

uniform spread of points representative of all parts of the Pareto set and hence yields a better model of the trade-off curve for the same effective computational cost.

Finally, Fig. 14 shows the Pareto surface obtained using NBI with stress in the left bar and total volume and stress in the right bar as objectives. The uniform stepsize δ on each component of β was chosen to be 0.1, and 66 NBI subproblems were solved of which nine failed to converge owing to infeasibility. The whole process took about 11.4 million flops.

A more detailed engineering-oriented treatment of this problem with trade-off studies for more than the groups of objectives mentioned here can be found in Das [1].

12. Function scaling implicit in NBI. NBI using the quasi-normal component is unaffected by the function scales. However, as the functions get more disparately scaled, the Pareto set gets more “stretched,” and consequently the NBI

TABLE 2

Weights (w_1, w_2)	Objective values (original scale)	Objective values (f_1 scaled by 5)	Objective values (f_1 scaled by 10)
0.00, 1.00	10.0000, -4.0111	10.0000, -4.0111	10.0000, -4.0111
0.05, 0.95	10.0000, -4.0111	10.0000, -4.0111	4.8211, -1.6330
0.10, 0.90	10.0000, -4.0111	4.1857, -1.2896	1.1634, 0.8741
0.15, 0.85	10.0000, -4.0111	1.6131, 0.4330	0.7689, 1.4083
0.20, 0.80	10.0000, -4.0111	1.0180, 1.0451	0.6559, 1.6416
0.25, 0.75	10.0000, -4.0111	0.7975, 1.3592	0.6100, 1.7724
0.30, 0.70	8.9403, -3.5644	0.6953, 1.5506	0.5876, 1.8563
0.35, 0.65	4.5379, -1.4822	0.6412, 1.6796	0.5754, 1.9146
0.40, 0.60	2.7307, -0.4109	0.6100, 1.7725	0.5682, 1.9576
0.45, 0.55	1.8319, 0.2473	0.5909, 1.8425	0.5637, 1.9905
0.50, 0.50	1.3357, 0.6928	0.5788, 1.8973	0.5608, 2.0165
0.55, 0.45	1.0425, 1.0147	0.5707, 1.9413	0.5589, 2.0376
0.60, 0.40	0.8615, 1.2583	0.5654, 1.9773	0.5576, 2.0551
0.65, 0.35	0.7463, 1.4492	0.5618, 2.0075	0.5567, 2.0698
0.70, 0.30	0.6719, 1.6029	0.5593, 2.0331	0.5561, 2.0823
0.75, 0.25	0.6236, 1.7295	0.5576, 2.0551	0.5557, 2.0931
0.80, 0.20	0.5926, 1.8356	0.5565, 2.0741	0.5554, 2.1025
0.85, 0.15	0.5734, 1.9258	0.5558, 2.0909	0.5553, 2.1108
0.90, 0.10	0.5622, 2.0035	0.5554, 2.1057	0.5552, 2.1181
0.95, 0.05	0.5567, 2.0711	0.5551, 2.1188	0.5551, 2.1247
1.00, 0.00	0.5551, 2.1306	0.5551, 2.1306	0.5551, 2.1306

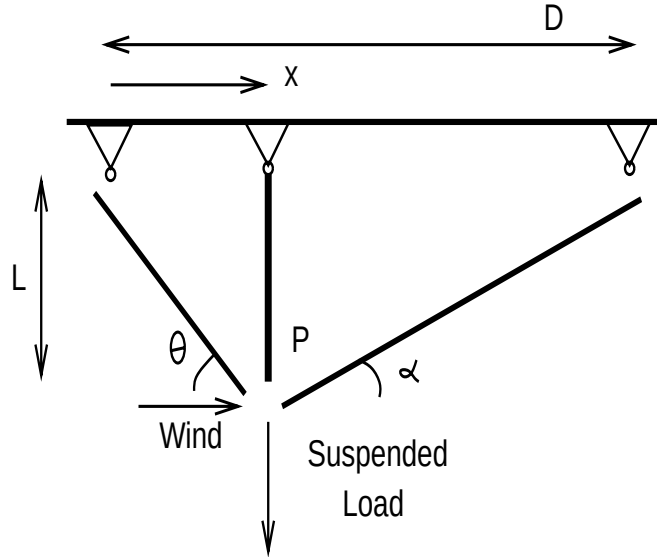


FIG. 11. A truss structure under a suspended load and a wind load.

points get farther apart from each other. Consequently solving an NBI subproblem starting from the solution of the same nearby subproblem takes more iterations to converge. This was observed in the numerical example above and motivates the need to scale the functions properly to remove this disparity in scales.

Geometrically it can be perceived that if the vertices of the *CHIM* simplex are almost equidistant from the origin, i.e., the quantities

$$\|F(x_j^*) - F^*\|, \quad j = 1, \dots, n,$$

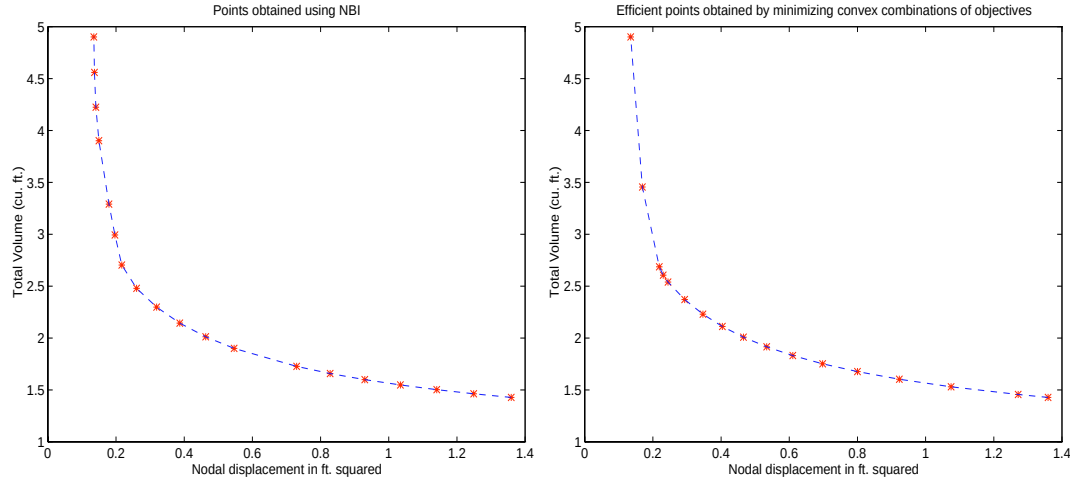


FIG. 12. Pareto curves for minimizing nodal displacement and total volume (cu. ft.) using NBI and convex combinations.

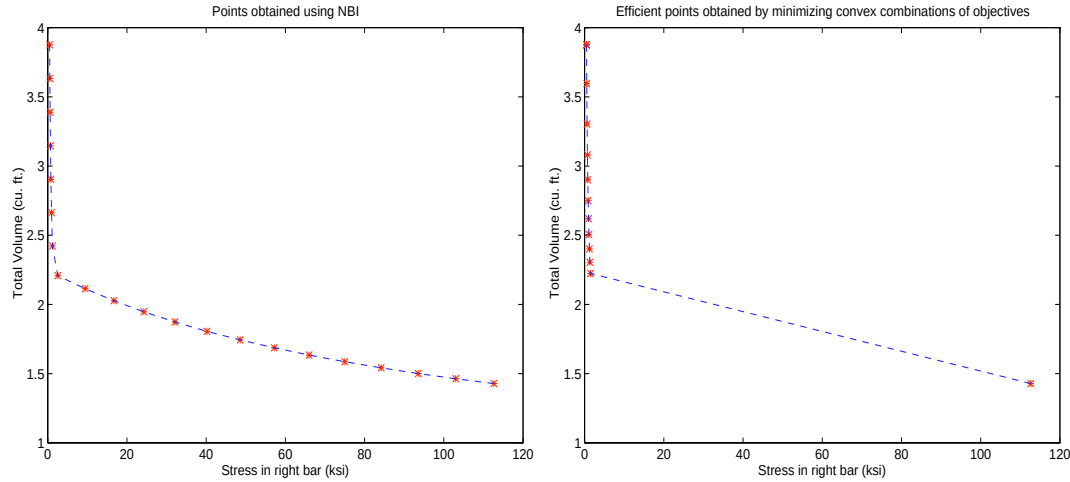


FIG. 13. NBI points for minimizing stress in right bar and total volume (cu. ft.).

are almost equal, then the quasi-normal direction $\hat{n}$ is almost normal to the $CHIM$ simplex. This would achieve the “minimally stretched” Pareto set we want and could also be a good scaling for the problem in the sense that all the functions would be about the same order of magnitude and thus would reduce possible ill conditioning.

For the biobjective problem, Φ is antidiagonal; thus a scaling that would achieve the above is obvious:

$$f_1 \leftarrow \frac{f_1}{f_1(x_2^*)},$$

$$f_2 \leftarrow \frac{f_2}{f_2(x_1^*)},$$

which makes each vertex of $CHIM$ be unit distance from the origin.

TABLE 3

	NBI	Convex combinations
Number of distinct points	21	12
Number of flops	2,962,068	1,650,079
Flops per distinct point	141,050.86	137,506.58

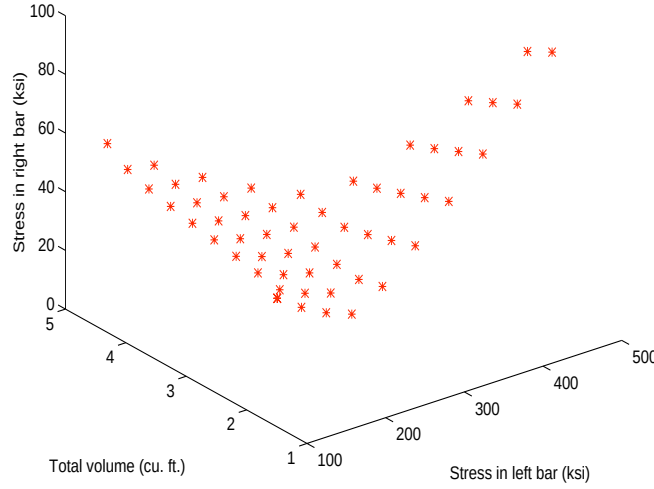


FIG. 14. NBI points for minimizing stresses in left and right bars and total volume.

However, the solution may not be so transparent for more than two objectives, and it may not be possible to get all the vertices exactly equidistant from the origin. So now we shall attempt to find function scalings $d_i > 0$ such that the functions scaled as

$$f_i \leftarrow \sqrt{d_i} f_i$$

will have the property that the *variance* among the scaled distances of the vertices from the origin, i.e.,

$$\|\sqrt{D}(F(x_j^*) - F^*)\|^2, \quad j = 1, \dots, n,$$

will be minimized ($D = \text{diag}(d)$; d represents the vector with components d_i).

Let $v_j = \|\sqrt{D}(F(x_j^*) - F^*)\|^2$, i.e.,

$$v_j = \sum_{i=1}^n d_i \phi_{i,j}^2,$$

where $\phi_{i,j}$ is the i th row j th column entry of the matrix Φ .

The mean square distance of the vertices is defined as

$$\bar{v} = \frac{1}{n} \sum_{j=1}^n v_j = \frac{1}{n} \sum_{i=1}^n d_i \left(\sum_{j=1}^n \phi_{i,j}^2 \right).$$

The variance quantity to be minimized is given by

$$V(d) = \sum_{j=1}^n (v_j - \bar{v})^2;$$

i.e.,

$$V(d) = \sum_{j=1}^n \left\{ \sum_{i=1}^n d_i \phi_{i,j}^2 - \sum_{i=1}^n d_i \left(\frac{1}{n} \sum_{j=1}^n \phi_{i,j}^2 \right) \right\}^2.$$

Let A be the matrix with components $a_{i,j}$ given by

$$a_{i,j} = \phi_{i,j}^2 - \frac{1}{n} \sum_{k=1}^n \phi_{i,k}^2.$$

Then

$$V(d) = \sum_{j=1}^n \left(\sum_{i=1}^n d_i a_{i,j} \right)^2;$$

i.e.,

$$V(d) = d^T A A^T d = \|A^T d\|^2.$$

This quadratic function is convex in d and has an unconstrained minimizer at $d = 0$. Thus we shall demand a specific value of $\bar{v}$, which represents an average distance of the CHIM simplex from the origin and is roughly the same order of magnitude as a typical function value of any objective encountered in the computation.

Suppose we want a typical objective value to be τ , which could be something like 10. Then we would enforce

$$\bar{v} = \frac{1}{n} \sum_{i=1}^n d_i \left(\sum_{j=1}^n \phi_{i,j}^2 \right) = \tau$$

along with a small lower bound on d_i . Thus the optimization problem to be solved to obtain our “optimal” scales is

$$\begin{aligned} \min_d V(d) &= d^T A A^T d \\ \text{s.t.} \quad &\sum_{i=1}^n d_i \left(\sum_{j=1}^n \phi_{i,j}^2 \right) = n\tau, \end{aligned}$$

$$d_i \geq 10^{-8}, \quad i = 1, \dots, n.$$

Thus we can see how the matrix Φ suggests an “improved scaling” of the objective functions, which is a bonus in the NBI approach.

It is worth observing that using the mean distance as opposed to the mean square distance in the last constraint would result in loss of convexity; hence the latter is preferred.

13. Conclusion. A technique was presented for finding Pareto optimal points of any smooth, constrained multiobjective problem with any number of objectives, perhaps restricted only by considerations of computational expense. The technique is efficient and has several useful properties, including that of obtaining an even spread of Pareto optimal points and invariance with respect to function scaling. This technique should be regarded as a tool for generating points from which the user can select the final design and not one that actually helps the user make that selection.

Further research is in progress regarding the implementational issues of parallelizing the solution of the NBI subproblem. Customized NLP techniques for solving the NBI subproblem will also be investigated.

A public domain MATLAB 4.2 implementation of NBI is available free of charge at <http://www.owl.net.rice.edu/~indra/NBIhomepage.html>.

Acknowledgments. First, the authors would like to thank the referees for pointing out various improvements in the presentation of the material and presenting several insightful comments. The authors would also like to thank Paul Uhlig, Department of Mathematics, Rice University, for several insightful discussions; Dr. Jagannatha Rao, Department of Mechanical Engineering, University of Houston, for providing motivation and helpful comments; Dan P. Giesy, NASA-Langley, for an interesting conversation on his earlier work; Jeffrey Hittinger, University of Michigan, Ann Arbor, for a helpful discussion on data structures; and Sanjeeb Dash, Rice University, for his algorithmic insights.

REFERENCES

- [1] I. DAS, *Nonlinear Multicriteria Optimization and Robust Optimality*, Ph.D. Thesis, Dept. of Computational and Applied Mathematics, Rice University, Houston, TX, 1997.
- [2] I. DAS AND J. E. DENNIS, *A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems*, Structural Optim., 14 (1997), pp. 63–69.
- [3] H. ESCHENAUER, J. KOSKI, AND A. OSYCZKA, *Multicriteria Design Optimization*, Springer-Verlag, Berlin, 1990.
- [4] R. B. STATNIKOV AND J. B. MATUSOV, *Multicriteria Optimization and Engineering*, Chapman and Hall, New York, 1995.
- [5] J. G. LIN, *Three methods for determining Pareto-optimal solutions of multiple-objective problems*, in Directions in Large-Scale Systems, Y. C. Ho and S. K. Mitter, eds., Plenum Press, New York, 1975, pp. 117–138.
- [6] J. RAKOWSKA, R. T. HAFTKA, AND L. T. WATSON, *Tracing the efficient curve for multi-objective control-structure optimization*, Comput. Systems Engrg., 2 (1991), pp. 461–471.
- [7] J. R. RAO AND P. Y. PAPALAMBROS, *A non-linear programming continuation strategy for one parameter design optimization problems*, in Proceedings of ASME Design Automation Conference, Montreal, Quebec, Canada, Sept. 17–20, 1989, pp. 77–89.
- [8] B. N. LUNDBERG AND A. B. POORE, *Bifurcations and sensitivity in parametric programming*, in Proceedings of Third Air Force/NASA Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, San Francisco, CA, Sept. 24–26, 1990, pp. 50–55.
- [9] F. W. GEMBICKI, *Performance and Sensitivity Optimization: A Vector Index Approach*, Ph.D. Thesis, Dept. of Systems Engineering, Case Western Reserve University, Cleveland, OH, 1974.
- [10] A. A. SCHY AND D. P. GIESY, *Tradeoff Studies in Multiobjective Insensitive Design of Airplane Control Systems*, in Proc. of the AIAA Guidance and Control Conference, Gatlinburg, TN, Aug. 15–17, 1983.
- [11] A. A. SCHY AND D. P. GIESY, *Multiobjective Insensitive Design of Airplane Control Systems with Uncertain Parameters*, in Proc. of the AIAA Guidance and Control Conference, Albuquerque, NM, Aug. 19–21, 1981.

- [12] A. A. SCHY, D. P. GIESY, AND K. J. JOHNSON, *Pareto-optimal multi-objective design of airplane control systems*, in Proceedings of the 1980 Joint Automatic Control Conference, American Automatic Control Council, ASME, Vol. 1, New York, 1980.
- [13] A. A. SCHY AND D. P. GIESY, *Multicriteria optimization methods for design of aircraft control systems*, in Multicriteria Optimization in Engineering and in the Sciences, W. Stadler, ed., Plenum Press, New York, 1988.
- [14] J. KOSKI, *Multicriteria truss optimization*, in Multicriteria Optimization in Engineering and in the Sciences, W. Stadler, ed., New York, Plenum Press, 1988.
- [15] Y. HAIMES, W. HALL, AND H. FREEDMAN, *Multiobjective Optimization in Water Resources Systems*, Elsevier Scientific Publishing Co., Amsterdam, 1975.
- [16] J. G. LIN, *Multiple-objective problems: Pareto-optimal solutions by method of proper equality constraints*, IEEE Trans. Automat. Control, AC-21 (1976), pp. 641–650.
- [17] G. A. KATOPIS AND J. G. LIN, *Non-inferiority of controls under double performance objectives: Minimal time and minimal energy*, in Proceedings of 7th Hawaii Int. Conf. Syst. Sci., Honolulu, HI, Jan. 1974, pp. 129–131.
- [18] J. G. LIN, *Circuit design under multiple performance objectives*, in Proceedings of the 1974 IEEE Int. Symp. Circuits and Systems, San Francisco, CA, Apr. 1974, pp. 549–552.

PARALLEL VARIABLE TRANSFORMATION IN UNCONSTRAINED OPTIMIZATION*

MASAO FUKUSHIMA†

Abstract. A general framework for unconstrained minimization of a nonlinear function using parallel processors is presented. The basic idea underlying the proposed parallel variable transformation algorithm is to transform the variables into more than one space of smaller dimension simultaneously and compute candidate solutions on the latter spaces in parallel. The candidate solutions obtained are then used to generate an improved solution to the original problem. Global convergence and the linear rate of convergence of the algorithm are established under suitable conditions. Two recently proposed parallel optimization algorithms, the parallel gradient distribution (PGD) algorithm and the unconstrained parallel variable distribution (PVD) algorithm, are shown to belong to the class of parallel variable transformation (PVT) algorithms. An earlier parallel algorithm called the updated conjugate subspaces (UCS) method is also shown to be a particular case of the PVT algorithm. Specific algorithmic schemes are also suggested.

Key words. parallel algorithm, unconstrained optimization, nonlinear programming, descent method

AMS subject classifications. 90C30, 49M37, 68Q22, 65Y05

PII. S1052623496309879

1. Introduction. The purpose of this paper is to present a general framework for designing parallel algorithms for the unconstrained minimization problem

$$(1) \quad \min_{x \in \mathbb{R}^n} f(x),$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is assumed to be continuously differentiable and bounded below.

Recently Ferris and Mangasarian [4] proposed a parallel algorithm for solving problem (1) based on the concept of PVD. In their algorithm, called the PVD algorithm, the variables are distributed among processors. However, the main difference from other classical methods such as the block Jacobi and coordinate descent methods [1] lies in the fact that each processor not only has primary responsibility for updating its own block of variables but also allows the other secondary variables to change in a restricted manner. With the help of this additional work, the PVD algorithm is expected to have enhanced practical efficiency. Assuming that each processor computes an exact solution of its subproblem, it was proved in [4] that any accumulation point of the generated sequence is stationary to the original problem. Also the linear rate of convergence was proved in [4] under the strong convexity assumption on the objective function. More recently, Solodov [8] succeeded in establishing the corresponding results by replacing these restrictive conditions with much weaker ones. Also Solodov [9] extended the algorithm of [4] to problems with general convex constraints.

The approach proposed in this paper is an extension of the methods presented in [4] and [8]. We no longer insist that a particular block of variables is assigned to each processor. Instead, we make use of multiple transformations of the variables into

* Received by the editors September 29, 1996; accepted for publication (in revised form) April 2, 1997; published electronically June 3, 1998.

<http://www.siam.org/journals/siopt/8-3/30987.html>

† Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan (fuku@kuamp.kyoto-u.ac.jp). The research of the author was supported in part by the Scientific Research Grant-in-Aid from the Ministry of Education, Science, Sports, and Culture, Japan.

spaces of smaller dimension in such a way that the latter spaces collectively span, in a certain sense, the space of the original variables. Therefore, in general, there is no distinction between the primary and the secondary variables for each processor. This approach will be called parallel variable transformation (PVT). At each iteration of the PVT algorithm, we compute multiple candidate solutions by solving subproblems on the transformed spaces in parallel. The candidate solutions obtained are then used to generate an improved solution to the original problem. A significant feature of the parallelization phase is that a single descent iteration for solving each subproblem may suffice to obtain the overall convergence.

The PVT algorithm is also closely related to another parallel unconstrained optimization algorithm proposed by Mangasarian [6], which is called the parallel gradient distribution (PGD) algorithm. In the PGD algorithm, the variables are distributed among processors like the PVD algorithm of [4]. However, unlike the PVD algorithm, each processor is responsible for updating the assigned variables only. Thus this algorithm is similar in spirit to the classical block Jacobi and coordinate descent methods. Nevertheless the PGD algorithm is significantly different from those methods in that each subproblem in the parallelization phase needs to be solved only approximately, for example, by executing a single descent iteration. In this respect, the PGD algorithm and the PVT algorithm proposed in this paper have a common feature. In fact, the PGD algorithm may be considered to belong to the class of the PVT algorithm. No convergence rate result is given for the PGD algorithm in [6], while the linear rate of convergence of the PVT algorithm is established in the present paper.

The PVT algorithm also has an intimate relation with the updated conjugate subspaces (UCS) method presented by Han [5]. The UCS method constructs subspaces that are mutually conjugate with respect to a symmetric positive definite matrix G . If G is chosen to be the Hessian $\nabla^2 f(x)$ of the objective function and the quadratic model function $q(v) \equiv f(x) + \nabla f(x)^\top v + \frac{1}{2}v^\top Gv$ is minimized over each subspace in parallel, then the aggregated solution yields the Newton direction for f at x . Particular attention is paid to the case where G is an approximate Hessian updated by the BFGS formula at every iteration, and the subspaces are also updated to maintain conjugacy with respect to the updated approximate Hessian matrix. A number of variants of the UCS method are suggested in [5] without rigorous convergence analysis. The UCS method has been applied to partially separable minimization problems in [2]. The iterated-subspace minimization (ISM) method of Conn et al. [3] also minimizes the objective function over a low-dimensional subspace constructed using conjugate directions at each iteration. However, the ISM method is sequential in itself and is considered a multidimensional search analogue of the ordinary (one-dimensional) line-search method.

This paper is organized as follows. In section 2, we formally state the PVT algorithm and establish global convergence of the algorithm under some basic assumptions. In section 3, we consider a general descent iteration scheme to solve subproblems in the parallelization phase and show that the basic assumptions are satisfied under suitable conditions on the transformations adopted. In section 4, we establish a linear rate of convergence of the PVT algorithm under some additional assumptions. In section 5, we demonstrate that the unconstrained PVD algorithm and the PGD algorithm are special instances of the PVT algorithm. We also relate the PVT algorithm to the UCS method. In section 6, we suggest some specific algorithms derived from the general framework of the PVT algorithm. In the final section, we give brief concluding remarks.

2. PVT. We present a prototype of the PVT algorithm for solving problem (1). We suppose that the algorithm is implemented on p processors, where p is an arbitrary positive integer. Each iteration of the algorithm consists of the parallelization phase and the synchronization phase. The former produces multiple candidate solutions for the next iterate in parallel using p processors, while the latter generates the next iterate from the candidate solutions obtained in the parallelization phase.

ALGORITHM PVT (PARALLEL VARIABLE TRANSFORMATION)

Step 0 (Initialization): Let p be the number of parallel processors. Let m_ℓ , $\ell = 1, \dots, p$ be positive integers such that $m_1 + \dots + m_p \geq n$. Choose an initial point $x^{(0)} \in \mathbb{R}^n$ and let $k := 0$.

Step 1 (Parallelization): For each $\ell \in \{1, \dots, p\}$, choose an $n \times m_\ell$ matrix $A_\ell^{(k)}$ and find an approximate solution $y_\ell^{(k)} \in \mathbb{R}^{m_\ell}$ to the minimization problem

$$(2) \quad \min_{y_\ell \in \mathbb{R}^{m_\ell}} \varphi_\ell^{(k)}(y_\ell) \equiv f(A_\ell^{(k)} y_\ell + x^{(k)}).$$

If $\nabla \varphi_\ell^{(k)}(0) = 0$ for all ℓ , then stop. Otherwise, go to Step 2.

Step 2 (Synchronization): Let $B^{(k)}$ be the $n \times (p+1)$ matrix having columns $x^{(k)}$ and $A_\ell^{(k)} y_\ell^{(k)} + x^{(k)}$, $\ell = 1, \dots, p$. Find an approximate solution $z^{(k)} = (z_0^{(k)}, z_1^{(k)}, \dots, z_p^{(k)})^\top \in \mathbb{R}^{p+1}$ to the minimization problem

$$(3) \quad \min_{z \in \mathbb{R}^{p+1}} \psi^{(k)}(z) \equiv f(B^{(k)} z).$$

Let $x^{(k+1)} := B^{(k)} z^{(k)}$ and $k := k + 1$. Return to Step 1.

We presume that m_ℓ , the number of variables in each subproblem (2), is significantly smaller than n , the number of variables in the original problem (1). Moreover, approximate solutions to problems (2) and (3) to be computed in Steps 1 and 2, respectively, are not required to be very accurate. In fact, for each ℓ and k , we may only require $y_\ell^{(k)}$ to satisfy

$$(4) \quad \varphi_\ell^{(k)}(y_\ell^{(k)}) \leq \varphi_\ell^{(k)}(0) - \sigma_\ell^{(k)}(\|\nabla \varphi_\ell^{(k)}(0)\|),$$

where $\sigma_\ell^{(k)} : \mathbb{R} \rightarrow \mathbb{R}$ is a forcing function associated with the function $\varphi_\ell^{(k)}$. Recall that a continuous function $\sigma : [0, \infty) \rightarrow [0, \infty)$ is called a forcing function [7] if for any nonnegative sequence $\{t_k\} \subset [0, \infty)$,

$$\sigma(t_k) \rightarrow 0 \implies t_k \rightarrow 0.$$

Note that, by definition,

$$(5) \quad \varphi_\ell^{(k)}(0) = f(x^{(k)})$$

and

$$(6) \quad \nabla \varphi_\ell^{(k)}(0) = (A_\ell^{(k)})^\top \nabla f(x^{(k)})$$

for all ℓ and k . Condition (4) will be satisfied, for example, by executing a single descent iteration for $\varphi_\ell^{(k)}$ with starting point $y_\ell = 0$ (see [7, Chapter 14] and [6, pp.

1919–1920]). Concerning the synchronization phase, for each k we may only require $z^{(k)}$ to satisfy

$$(7) \quad \psi^{(k)}(z^{(k)}) \leq \min_{1 \leq \ell \leq p} \varphi_\ell^{(k)}(y_\ell^{(k)}).$$

This condition is trivially satisfied if we simply choose $z^{(k)}$ to be the unit vector such that $B^{(k)}z^{(k)} = A_{\bar{\ell}}^{(k)}y_{\bar{\ell}}^{(k)} + x^{(k)}$, where $\bar{\ell}$ is an index corresponding to the minimum on the right-hand side of (7). In other words, we may choose $x^{(k+1)}$ to be the best of the candidate points $A_{\ell}^{(k)}y_{\ell}^{(k)} + x^{(k)}$, $\ell = 1, \dots, p$.

To establish global convergence of the PVT algorithm, we need the following assumptions:

(A1) Let $A^{(k)} := [A_1^{(k)} \dots A_p^{(k)}] \in \mathbb{R}^{n \times (m_1 + \dots + m_p)}$. There exists a constant $\beta > 0$ independent of k such that, for all k ,

$$(8) \quad \|(A^{(k)})^\top x\| \geq \beta \|x\| \quad \text{for all } x \in \mathbb{R}^n.$$

(A2) For each ℓ , the sequence $\{\sigma_\ell^{(k)} \mid k = 0, 1, 2, \dots\}$ of forcing functions in (4) satisfies the property that, for any $\varepsilon > 0$, there exists a $\delta > 0$ independent of k such that for all k ,

$$\sigma_\ell^{(k)}(t) < \delta \implies t < \varepsilon.$$

The first assumption (A1) is equivalent to saying that the sequence $\{A^{(k)}(A^{(k)})^\top\}$ of $n \times n$ matrices is uniformly positive definite; i.e., there exists a constant $\hat{\beta} > 0$ independent of k such that

$$x^\top A^{(k)}(A^{(k)})^\top x \geq \hat{\beta} \|x\|^2 \quad \text{for all } x \in \mathbb{R}^n.$$

Roughly, assumption (A1) implies that each variable needs to be taken into account at each iteration. Failure of this assumption may lead to nonconvergence of the algorithm, as the following simple example shows. Consider the problem

$$\min_{x \in \mathbb{R}^2} x_1^2 + (x_2 - 1)^2,$$

which has the unique minimizer $x = (0, 1)^\top$. Let $p = 2$, $m_1 = 1$, and $m_2 = 1$. Choose the initial point $x^{(0)} = (0, 0)^\top$. Choose the transformation matrices as

$$A_1^{(0)} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad A_2^{(0)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Then the PVT algorithm immediately terminates at $x^{(0)}$, which is not a stationary point of the problem. Therefore, care has to be taken in choosing the transformation matrices to prevent stalling at a nonstationary point.

The second condition (A2) requires a “uniformly forcing property” of forcing functions $\sigma_\ell^{(k)}$ associated with the functions $\varphi_\ell^{(k)}$. This key condition will be discussed in detail in the next section. Note that, under condition (A2), the function $\sigma_\ell : [0, \infty) \rightarrow [0, \infty)$ defined by

$$(9) \quad \sigma_\ell(t) \equiv \inf_k \sigma_\ell^{(k)}(t)$$

also satisfies the property of a forcing function.

THEOREM 2.1. *Suppose that, at each iteration of the PVT algorithm, conditions (4) and (7) are satisfied in Steps 1 and 2, respectively. Assume in addition that conditions (A1) and (A2) are satisfied. Then the PVT algorithm either terminates at a stationary point of problem (1) or generates an infinite sequence $\{x^{(k)}\}$ such that*

$$\lim_{k \rightarrow \infty} \nabla f(x^{(k)}) = 0$$

and every accumulation point of $\{x^{(k)}\}$ is a stationary point of problem (1).

Proof. First suppose that the algorithm terminates, say, at iteration k . Then we have $\nabla \varphi_\ell^{(k)}(0) = 0$ for all ℓ , which implies by (6)

$$(A_\ell^{(k)})^\top \nabla f(x^{(k)}) = 0, \quad \ell = 1, \dots, p.$$

Since $A^{(k)} = [A_1^{(k)} \cdots A_p^{(k)}]$, we get

$$\begin{aligned} \|(A^{(k)})^\top \nabla f(x^{(k)})\|^2 &= \sum_{\ell=1}^p \|(A_\ell^{(k)})^\top \nabla f(x^{(k)})\|^2 \\ &= 0. \end{aligned}$$

It then follows from (8) that

$$\nabla f(x^{(k)}) = 0.$$

Thus $x^{(k)}$ is a stationary point of (1).

Next suppose that the algorithm generates an infinite sequence $\{x^{(k)}\}$. Since $x^{(k+1)} = B^{(k)} z^{(k)}$, it follows from the definitions of $\psi^{(k)}$ and $\varphi_\ell^{(k)}$ that

$$(10) \quad f(x^{(k)}) - f(x^{(k+1)}) = \varphi_\ell^{(k)}(0) - \psi^{(k)}(z^{(k)}), \quad \ell = 1, \dots, p.$$

Moreover, by (4), we have

$$(11) \quad \begin{aligned} \varphi_\ell^{(k)}(0) &\geq \varphi_\ell^{(k)}(y_\ell^{(k)}) + \sigma_\ell^{(k)}(\|\nabla \varphi_\ell^{(k)}(0)\|) \\ &\geq \varphi_\ell^{(k)}(y_\ell^{(k)}) + \sigma_\ell(\|\nabla \varphi_\ell^{(k)}(0)\|), \quad \ell = 1, \dots, p, \end{aligned}$$

where σ_ℓ is given by (9). Since (7) holds, (10) and (11) imply that

$$(12) \quad f(x^{(k)}) - f(x^{(k+1)}) \geq \sigma_\ell(\|\nabla \varphi_\ell^{(k)}(0)\|) \geq 0, \quad \ell = 1, \dots, p.$$

Thus $\{f(x^{(k)})\}$ is nonincreasing. Moreover, since f is bounded below by our blanket assumption, (12) implies that

$$\lim_{k \rightarrow \infty} f(x^{(k)}) - f(x^{(k+1)}) = 0,$$

and hence

$$(13) \quad \lim_{k \rightarrow \infty} \sigma_\ell(\|\nabla \varphi_\ell^{(k)}(0)\|) = 0.$$

Since σ_ℓ is a forcing function under condition (A2), it follows from (6) and (13) that

$$(14) \quad \lim_{k \rightarrow \infty} \|(A_\ell^{(k)})^\top \nabla f(x^{(k)})\| = 0, \quad \ell = 1, \dots, p.$$

On the other hand, noting that $A^{(k)} = [A_1^{(k)} \cdots A_p^{(k)}]$, we obtain from (8) that

$$\begin{aligned} \sum_{\ell=1}^p \|(A_\ell^{(k)})^\top \nabla f(x^{(k)})\|^2 &= \|(A^{(k)})^\top \nabla f(x^{(k)})\|^2 \\ &\geq \beta \|\nabla f(x^{(k)})\|^2. \end{aligned}$$

Therefore, (14) implies that

$$\lim_{k \rightarrow \infty} \nabla f(x^{(k)}) = 0.$$

Consequently, any accumulation point of $\{x^{(k)}\}$ is a stationary point of (1). $\square$

3. Uniformly forcing property. As demonstrated in the previous section, we need not compute accurate approximate solutions to subproblems (2) at every iteration of the PVT algorithm. More specifically, at every iteration k , it is sufficient to find $y_\ell^{(k)}$ such that condition (4) is satisfied for $\ell = 1, \dots, p$. This is not a difficult task, because condition (4) will be satisfied if we execute a single descent iteration to minimize $\varphi_\ell^{(k)}$ starting at the origin $y_\ell = 0$. In the following, we suppose that $y_\ell^{(k)}$ is actually determined by

$$(15) \quad y_\ell^{(k)} = \alpha_\ell^{(k)} d_\ell^{(k)},$$

where $d_\ell^{(k)}$ is a descent direction satisfying a gradient-relatedness condition [7]

$$(16) \quad \nabla \varphi_\ell^{(k)}(0)^\top d_\ell^{(k)} \leq -\mu_0 \|\nabla \varphi_\ell^{(k)}(0)\| \|d_\ell^{(k)}\|$$

with some constant $\mu_0 > 0$, and the stepsize $\alpha_\ell^{(k)} > 0$ is chosen according to the Goldstein rule

$$(17) \quad \varphi_\ell^{(k)}(\alpha_\ell^{(k)} d_\ell^{(k)}) \leq \varphi_\ell^{(k)}(0) + \mu_1 \alpha_\ell^{(k)} \nabla \varphi_\ell^{(k)}(0)^\top d_\ell^{(k)},$$

$$(18) \quad \varphi_\ell^{(k)}(\alpha_\ell^{(k)} d_\ell^{(k)}) \geq \varphi_\ell^{(k)}(0) + \mu_2 \alpha_\ell^{(k)} \nabla \varphi_\ell^{(k)}(0)^\top d_\ell^{(k)},$$

with constants μ_1 and μ_2 such that $0 < \mu_1 \leq \mu_2 < 1$. We assume that μ_0, μ_1 , and μ_2 are independent of ℓ and k . Condition (16) holds if we determine $d_\ell^{(k)}$ as

$$d_\ell^{(k)} = -H_\ell^{(k)} \nabla \varphi_\ell^{(k)}(0),$$

where $H_\ell^{(k)} \in \Re^{m_\ell \times m_\ell}$ are symmetric positive definite matrices such that

$$\|H_\ell^{(k)}\| \leq \mu_3 \quad \text{and} \quad d^\top H_\ell^{(k)} d \geq \mu_4 \|d\|^2 \quad \text{for all } d \in \Re^{m_\ell}$$

for some constants $\mu_3 \geq \mu_4 > 0$ independent of ℓ and k (see [7, Theorem 14.4.1]).

Recall that Theorem 2.1 requires, among others, the uniformly forcing property (A2) for forcing functions $\sigma_\ell^{(k)}, k = 1, 2, \dots$, which appear in the inequalities (4). To verify this key condition, it suffices to show the existence of a nondecreasing forcing function $\hat{\sigma}_\ell : [0, \infty) \rightarrow [0, \infty)$ for each ℓ which is independent of k and satisfies

$$(19) \quad \varphi_\ell^{(k)}(y_\ell^{(k)}) \leq \varphi_\ell^{(k)}(0) - \hat{\sigma}_\ell(\|\nabla \varphi_\ell^{(k)}(0)\|) \quad \text{for all } k = 0, 1, 2, \dots$$

In the next lemma, we show that the existence of such a function is guaranteed under the boundedness assumption on the sequence of matrices $\{A_\ell^{(k)} \mid k = 0, 1, 2, \dots\}$:

(A3) For each ℓ , there exists a constant $\lambda_\ell > 0$ such that

$$(20) \quad \|A_\ell^{(k)}\| \leq \lambda_\ell \quad \text{for all } k = 0, 1, 2, \dots$$

For an arbitrary starting point $x^{(0)} \in \mathfrak{R}^n$, we denote the level set of f by

$$\mathcal{L}(x^{(0)}) \equiv \{x \in \mathfrak{R}^n \mid f(x) \leq f(x^{(0)})\}$$

and the convex hull of $\mathcal{L}(x^{(0)})$ by $\bar{\mathcal{L}}(x^{(0)})$.

LEMMA 3.1. *Assume that either the level set $\mathcal{L}(x^{(0)})$ is compact or $\nabla f : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$ is Lipschitz continuous on the convex set $\bar{\mathcal{L}}(x^{(0)})$. Let $y_\ell^{(k)}$ be determined by the descent step (15) with (16)–(18). If condition (A3) is satisfied, then there exists a nondecreasing forcing function $\hat{\sigma}_\ell : [0, \infty) \rightarrow [0, \infty)$ for each ℓ such that the inequalities (19) hold.*

Proof. First consider the case where $\mathcal{L}(x^{(0)})$ is compact. Let $\omega_\ell : [0, \infty) \rightarrow [0, \infty)$ be the modulus of continuity of ∇f on $\bar{\mathcal{L}}(x^{(0)})$, which is defined by

$$(21) \quad \omega(t) \equiv \sup\{\|\nabla f(x) - \nabla f(x')\| \mid \|x - x'\| \leq t, x, x' \in \bar{\mathcal{L}}(x^{(0)})\}.$$

For each ℓ and k , let $D_\ell^{(k)}$ denote the convex set given by $D_\ell^{(k)} \equiv \{y_\ell \in \mathfrak{R}^{m_\ell} \mid A_\ell^{(k)} y_\ell + x^{(k)} \in \bar{\mathcal{L}}(x^{(0)})\}$ and define the function $\omega_\ell^{(k)}$ by

$$(22) \quad \omega_\ell^{(k)}(t) \equiv \sup\{\|\nabla \varphi_\ell^{(k)}(y_\ell) - \nabla \varphi_\ell^{(k)}(y'_\ell)\| \mid \|y_\ell - y'_\ell\| \leq t, y_\ell, y'_\ell \in D_\ell^{(k)}\}.$$

Since

$$\nabla \varphi_\ell^{(k)}(y_\ell) = (A_\ell^{(k)})^\top \nabla f(A_\ell^{(k)} y_\ell + x^{(k)}),$$

and since $A_\ell^{(k)} y_\ell + x^{(k)}$ lies in the compact set $\bar{\mathcal{L}}(x^{(0)})$ for each $y_\ell \in D_\ell^{(k)}$, the supremum on the right-hand side of (22) is attained by, say, $y_\ell \in D_\ell^{(k)}$ and $y'_\ell \in D_\ell^{(k)}$. Then we have

$$\begin{aligned} \omega_\ell^{(k)}(t) &= \|\nabla \varphi_\ell^{(k)}(y_\ell) - \nabla \varphi_\ell^{(k)}(y'_\ell)\| \\ &= \|(A_\ell^{(k)})^\top (\nabla f(A_\ell^{(k)} y_\ell + x^{(k)}) - \nabla f(A_\ell^{(k)} y'_\ell + x^{(k)}))\| \\ (23) \quad &\leq \lambda_\ell \|\nabla f(A_\ell^{(k)} y_\ell + x^{(k)}) - \nabla f(A_\ell^{(k)} y'_\ell + x^{(k)})\|, \end{aligned}$$

where the inequality follows from (20). Moreover we have

$$\begin{aligned} \|(A_\ell^{(k)} y_\ell + x^{(k)}) - (A_\ell^{(k)} y'_\ell + x^{(k)})\| &= \|A_\ell^{(k)}(y_\ell - y'_\ell)\| \\ &\leq \lambda_\ell \|y_\ell - y'_\ell\| \\ (24) \quad &\leq \lambda_\ell t. \end{aligned}$$

Therefore, the definition (21) of ω , (23), and (24) imply

$$(25) \quad \omega_\ell^{(k)}(t) \leq \lambda_\ell \omega(\lambda_\ell t) \quad \text{for all } k = 1, 2, \dots$$

The remainder of the proof is a modification of that of Theorem 14.2.14 in [7]. Define the functions $r : [0, \infty) \rightarrow [0, \infty)$ and $r_\ell^{(k)} : [0, \infty) \rightarrow [0, \infty)$ by

$$r(s) \equiv \int_0^1 \omega(st) dt$$

and

$$r_\ell^{(k)}(s) \equiv \int_0^1 \omega_\ell^{(k)}(st) dt,$$

respectively. From (25), we easily see that

$$(26) \quad r_\ell^{(k)}(s) \leq \lambda_\ell r(\lambda_\ell s) \quad \text{for all } k = 1, 2, \dots$$

Then we have

$$(27) \quad \begin{aligned} \varphi_\ell^{(k)}(\alpha_\ell^{(k)} d_\ell^{(k)}) &\leq \varphi_\ell^{(k)}(0) + \alpha_\ell^{(k)} \nabla \varphi_\ell^{(k)}(0)^\top d_\ell^{(k)} + \alpha_\ell^{(k)} \|d_\ell^{(k)}\| r_\ell^{(k)}(\alpha_\ell^{(k)} \|d_\ell^{(k)}\|) \\ &\leq \varphi_\ell^{(k)}(0) + \alpha_\ell^{(k)} \nabla \varphi_\ell^{(k)}(0)^\top d_\ell^{(k)} + \lambda_\ell \alpha_\ell^{(k)} \|d_\ell^{(k)}\| r(\lambda_\ell \alpha_\ell^{(k)} \|d_\ell^{(k)}\|), \end{aligned}$$

where the first and the second inequalities follow from Theorem 8.3.1 in [7] and (26), respectively. Combining (18) and (27) yields

$$-(1 - \mu_2) \nabla \varphi_\ell^{(k)}(0)^\top d_\ell^{(k)} \leq \lambda_\ell \|d_\ell^{(k)}\| r(\lambda_\ell \alpha_\ell^{(k)} \|d_\ell^{(k)}\|),$$

which together with (16) in turn implies that

$$(28) \quad \lambda_\ell^{-1} \mu_0 (1 - \mu_2) \|\nabla \varphi_\ell^{(k)}(0)\| \leq r(\lambda_\ell \alpha_\ell^{(k)} \|d_\ell^{(k)}\|).$$

Now let $\hat{r} : [0, \infty) \rightarrow [0, \infty)$ be any continuous, strictly increasing function such that $\hat{r}(0) = 0$ and $\hat{r}(t) \geq r(t)$ for all $t \geq 0$. Then by (28) we get

$$(29) \quad \lambda_\ell \alpha_\ell^{(k)} \|d_\ell^{(k)}\| \geq \hat{r}^{-1}(\lambda_\ell^{-1} \mu_0 (1 - \mu_2) \|\nabla \varphi_\ell^{(k)}(0)\|),$$

where $\hat{r}^{-1}$ is the inverse function of $\hat{r}$. So it follows that

$$(30) \quad \begin{aligned} \varphi_\ell^{(k)}(\alpha_\ell^{(k)} d_\ell^{(k)}) &\leq \varphi_\ell^{(k)}(0) - \mu_0 \mu_1 \alpha_\ell^{(k)} \|\nabla \varphi_\ell^{(k)}(0)\| \|d_\ell^{(k)}\| \\ &\leq \varphi_\ell^{(k)}(0) - \mu_0 \mu_1 \lambda_\ell^{-1} \|\nabla \varphi_\ell^{(k)}(0)\| \hat{r}^{-1}(\lambda_\ell^{-1} \mu_0 (1 - \mu_2) \|\nabla \varphi_\ell^{(k)}(0)\|), \end{aligned}$$

where the first inequality follows from (16) and (17) and the second inequality follows from (29). Define the function $\hat{\sigma}_\ell : [0, \infty) \rightarrow [0, \infty)$ by

$$\hat{\sigma}_\ell(t) = \mu_0 \mu_1 \lambda_\ell^{-1} t \hat{r}^{-1}(\lambda_\ell^{-1} \mu_0 (1 - \mu_2) t).$$

Note that $\hat{\sigma}_\ell$ enjoys the property of a forcing function. Then by (30) we have

$$(31) \quad \varphi_\ell^{(k)}(\alpha_\ell^{(k)} d_\ell^{(k)}) \leq \varphi_\ell^{(k)}(0) - \hat{\sigma}_\ell(\|\nabla \varphi_\ell^{(k)}(0)\|).$$

Consequently, by (15), we see that (19) is satisfied.

Finally consider the case where ∇f is Lipschitz continuous on the set $\bar{\mathcal{L}}(x^{(0)})$. In this case, there exists a constant $\kappa > 0$ such that

$$\omega(t) \leq \kappa t$$

for all $t \in [0, \infty)$. It then follows from (23) and (24) that

$$\omega_\ell^{(k)}(t) \leq \kappa \lambda_\ell^2 t \quad \text{for all } k = 1, 2, \dots$$

By specializing the preceding arguments used to derive (31), we can easily demonstrate that (19) holds with $\hat{\sigma}_\ell(t) = \delta_\ell t^2$ for some constant $\delta_\ell > 0$. $\square$

Combining the above result with the global convergence result established in the previous section, we obtain the next theorem.

THEOREM 3.1. *Assume that either the level set $\mathcal{L}(x^{(0)})$ is compact or $\nabla f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is Lipschitz continuous on the convex set $\tilde{\mathcal{L}}(x^{(0)})$. Suppose that, at each iteration of the PVT algorithm, $y_\ell^{(k)}$ are given by the descent step (15) with (16)–(18) and $z^{(k)}$ is determined to satisfy (3). Suppose also that the matrices $A_\ell^{(k)}$ are chosen to satisfy conditions (A1) and (A3). Then the PVT algorithm either terminates at a stationary point of problem (1) or generates an infinite sequence $\{x^{(k)}\}$ such that*

$$\lim_{k \rightarrow \infty} \nabla f(x^{(k)}) = 0$$

and every accumulation point of $\{x^{(k)}\}$ is a stationary point of problem (1).

Proof. By Lemma 3.1, there is a forcing function $\hat{\sigma}_\ell$ satisfying (19). This implies that condition (A2) is satisfied. Hence by Theorem 2.1, the assertion of the present theorem is verified. $\square$

We have supposed so far that, in Step 1 of the PVT algorithm, only one descent iteration is executed for each ℓ to minimize the function $\varphi_\ell^{(k)}$. It is clear, however, that the key condition (19) for Step 1 remains valid if more than one descent iteration for $\varphi_\ell^{(k)}$ is executed at any iteration k . On the other hand, condition (7) for Step 2 is easily satisfied either by simply choosing $z^{(k)}$ to be a point that corresponds to one of the $y_\ell^{(k)}$ with the lowest value of $\varphi_\ell^{(k)}$ or by executing an arbitrary number of descent iterations to minimize $\psi^{(k)}$ starting at any point corresponding to the lowest value of $\varphi_\ell^{(k)}(y_\ell^{(k)})$. We emphasize here that these facts allow us to implement the algorithm in a very flexible manner.

4. Rate of convergence. In this section, we discuss the rate of convergence of the algorithm presented in section 2. We assume that the generated sequence $\{x^{(k)}\}$ approaches the set S^* of minima of f , i.e.,

$$(32) \quad \lim_{k \rightarrow \infty} \text{dist}[x^{(k)}, S^*] = 0,$$

where $\text{dist}[x^{(k)}, S^*] \equiv \inf\{\|x^{(k)} - x\| \mid x \in S^*\}$. We impose the following conditions on the objective function f .

(B1) There exist constants $\rho > 0$, $c_1 > 0$, and $c_2 > 0$ such that $\text{dist}[x, S^*] \leq \rho$ implies that

$$(33) \quad \|\nabla f(x)\| \geq c_1 \text{dist}[x, S^*]$$

and

$$(34) \quad f(x) - f^* \leq c_2 \text{dist}[x, S^*]^2,$$

where $f^* \equiv \min_{x \in \mathbb{R}^n} f(x)$.

Similar conditions have been adopted by Solodov [9, Theorem 2.2] for the rate of convergence analysis of the constrained PVD algorithm. Note that (33) particularly holds if f is strongly convex around the minimum of f , in which case S^* reduces to a singleton. Condition (34) is satisfied if ∇f is Lipschitz continuous around S^* . Moreover, the subsequent argument will remain valid even if the set S^* is regarded as the set of local minima of f .

Concerning the algorithm, we assume the following condition, which is slightly more restrictive than condition (A2) given in section 2.

(B2) For each ℓ , there exists a constant $\delta_\ell > 0$ such that

$$(35) \quad \varphi_\ell^{(k)}(y_\ell^{(k)}) \leq \varphi_\ell^{(k)}(0) - \delta_\ell \|\nabla \varphi_\ell^{(k)}(0)\|^2 \quad \text{for all } k = 0, 1, 2, \dots$$

The proof of Lemma 3.1 indicates that, when $y_\ell^{(k)}$ is determined by the descent step (15) with (16)–(18), condition (B2) is fulfilled if condition (A3) is satisfied and ∇f is Lipschitz continuous on the set $\mathcal{L}(x^{(0)})$.

Now we state the rate of convergence result for the PVT algorithm.

THEOREM 4.1. *Assume that the sequence $\{x^{(k)}\}$ generated by the PVT algorithm satisfies (32) and the objective function f satisfies conditions (33) and (34). Moreover suppose that $y_\ell^{(k)}$ is chosen to satisfy condition (B2) in Step 1, $z^{(k)}$ is chosen to satisfy condition (7) in Step 2, and the matrices $A_\ell^{(k)}$ satisfy condition (A1). Then $\{f(x^{(k)})\}$ converges Q -linearly to f^* and $\{\text{dist}[x^{(k)}, S^*]\}$ converges R -linearly to zero.*

Proof. Since $f(x^{(k)}) = \varphi_\ell^{(k)}(0)$ for any ℓ and $f(x^{(k+1)}) = \psi^{(k)}(z^{(k)})$, it follows from (35), (7), and (6) that

$$\begin{aligned} f(x^{(k+1)}) &\leq f(x^{(k)}) - \delta_\ell \|(A_\ell^{(k)})^\top \nabla f(x^{(k)})\|^2 \\ &\leq f(x^{(k)}) - \delta \|(A_\ell^{(k)})^\top \nabla f(x^{(k)})\|^2, \quad \ell = 1, \dots, p, \end{aligned}$$

where $\delta \equiv \min_{1 \leq \ell \leq p} \delta_\ell > 0$. Hence we have

$$\begin{aligned} f(x^{(k)}) - f(x^{(k+1)}) &\geq \frac{\delta}{p} \sum_{\ell=1}^p \|(A_\ell^{(k)})^\top \nabla f(x^{(k)})\|^2 \\ &\geq \frac{\delta}{p} \|(A^{(k)})^\top \nabla f(x^{(k)})\|^2 \\ (36) \quad &\geq \frac{\delta \beta^2}{p} \|\nabla f(x^{(k)})\|^2, \end{aligned}$$

where the last inequality follows from (8) in condition (A1). Combining (36) with (33) and (34), we obtain

$$f(x^{(k)}) - f(x^{(k+1)}) \geq \frac{\delta \beta^2 c_1^2}{p c_2} (f(x^{(k)}) - f^*),$$

which in turn implies that

$$(37) \quad f(x^{(k+1)}) - f^* \leq c (f(x^{(k)}) - f^*),$$

where $c \equiv 1 - (\delta \beta^2 c_1^2) / (p c_2) \in (0, 1)$. Thus $\{f(x^{(k)})\}$ converges Q -linearly to f^* . Since (36) and (37) imply that

$$f(x^{(k)}) - f^* \geq \frac{\delta \beta^2}{p} \|\nabla f(x^{(k)})\|^2$$

and

$$f(x^{(k)}) - f^* \leq c^k (f(x^{(0)}) - f^*),$$

respectively, we obtain

$$\|\nabla f(x^{(k)})\|^2 \leq \frac{c^k p}{\delta \beta^2} (f(x^{(0)}) - f^*).$$

Hence the last half of the theorem follows from (33). $\square$

5. Related methods. The PVT Algorithm can take concrete shape diversely, depending upon specific choice of matrices $A_\ell^{(k)}$. In this section, we show that the PVT algorithm particularly contains the two recently proposed parallel optimization methods: the PGD algorithm of Mangasarian [6] and the unconstrained version of the PVD algorithm proposed by Ferris and Mangasarian [4] and further studied by Solodov [8]. We also relate the PVT algorithm to the UCS method of Han [5].

In the PGD algorithm, the vector $x \in \mathbb{R}^n$ of variables is decomposed into p subvectors $x_\ell \in \mathbb{R}^{n_\ell}$, $\ell = 1, \dots, p$, such that $n_1 + \dots + n_p = n$, i.e., $x = (x_1, \dots, x_p)$. At each iteration, given the current iterate $x^{(k)} \in \mathbb{R}^n$, the algorithm chooses for each ℓ a descent direction $d_\ell^{(k)} \in \mathbb{R}^{n_\ell}$ for the function $f(x_\ell, x_{\bar{\ell}}^{(k)})$ with $x_{\bar{\ell}}^{(k)}$ fixed, where $\bar{\ell}$ denotes the complement of ℓ in $\{1, \dots, p\}$ and $x_{\bar{\ell}}^{(k)}$ is the $(n - n_\ell)$ -dimensional subvector of $x^{(k)}$ composed of $x_i^{(k)}$, $i \in \bar{\ell}$. Using the directions $d_\ell^{(k)}$, the algorithm then finds, in parallel for each ℓ , a stepsize $\alpha_\ell^{(k)}$ by line search on the function $f(x_\ell, x_{\bar{\ell}}^{(k)})$ to produce p candidate points $(x_\ell^{(k)} + \alpha_\ell^{(k)} d_\ell^{(k)}, x_{\bar{\ell}}^{(k)})$, $\ell = 1, \dots, p$. Finally the next iterate $x^{(k+1)} \in \mathbb{R}^n$ is determined by taking the best of these candidate points or a better point. In the case where f is convex, the next iterate may alternatively be chosen as a convex combination of the candidate points.

Comparing PGD with PVT, we see that the latter is reduced to the former by letting, for each ℓ ,

$$\begin{aligned} m_\ell &:= n_\ell, \\ A_\ell^{(k)} &:= [0 \cdots 0 \ I_{n_\ell} \ 0 \cdots 0]^\top \in \mathbb{R}^{n \times n_\ell}, \end{aligned}$$

where the $n_\ell \times n_\ell$ identity matrix I_{n_ℓ} appears in the position corresponding to subvector x_ℓ . Then the resultant PVT is regarded as PGD through the identification

$$y_\ell = x_\ell - x_\ell^{(k)}, \quad \ell = 1, \dots, p.$$

Note that, in the PGD algorithm, matrix $A^{(k)} = [A_1^{(k)} \cdots A_p^{(k)}]$ becomes the $n \times n$ identity matrix and hence condition (A1) is trivially satisfied. Moreover, since $A_\ell^{(k)}$ does not depend on k , condition (A3) clearly holds, too.

The PVD algorithm presented by Ferris and Mangasarian [4] also decomposes the variables as $x = (x_1, \dots, x_p)$ with $x_\ell \in \mathbb{R}^{n_\ell}$, $\ell = 1, \dots, p$, and $n_1 + \dots + n_p = n$. Given the current iterate $x^{(k)} \in \mathbb{R}^n$, the next iterate $x^{(k+1)}$ is determined as follows: choose an arbitrary direction $d^{(k)} \in \mathbb{R}^n$, of which a particular choice is $d^{(k)} = -H^{(k)} \nabla f(x^{(k)})$ with a symmetric positive definite matrix $H^{(k)} \in \mathbb{R}^{n \times n}$. Then solve the following minimization problems in parallel for $\ell = 1, \dots, p$:

$$(38) \quad \min_{(x_\ell, \xi_{\bar{\ell}}) \in \mathbb{R}^{n_\ell + p-1}} \theta_\ell^{(k)}(x_\ell, \xi_{\bar{\ell}}) \equiv f(x_\ell, D_{\bar{\ell}}^{(k)} \xi_{\bar{\ell}} + x_{\bar{\ell}}^{(k)}),$$

where $\xi_{\bar{\ell}} := (\xi_1, \dots, \xi_{\ell-1}, \xi_{\ell+1}, \dots, \xi_p)^\top \in \mathbb{R}^{p-1}$ and

$$D_{\bar{\ell}}^{(k)} := \begin{bmatrix} d_1^{(k)} & & & & \\ & \ddots & & & \\ & & d_{\ell-1}^{(k)} & & \\ & & & d_{\ell+1}^{(k)} & \\ & & & & \ddots \\ & & & & & d_p^{(k)} \end{bmatrix} \in \mathbb{R}^{(n-n_\ell) \times (p-1)}.$$

For each ℓ , let $(\tilde{x}_\ell, \tilde{\xi}_\ell) \in \mathbb{R}^{n_\ell+p-1}$ be an optimal solution to subproblem (38). Let $\tilde{x}^{(k,\ell)} := (\tilde{x}_\ell, D_\ell^{(k)} \tilde{\xi}_\ell + x_\ell^{(k)}) \in \mathbb{R}^n$, $\ell = 1, \dots, p$, be the candidate solutions determined from $(\tilde{x}_\ell, \tilde{\xi}_\ell)$, $\ell = 1, \dots, p$, respectively. The next iterate $x^{(k+1)}$ is then sought by minimizing the objective function f over the affine hull of $p+1$ points $x^{(k)}, \tilde{x}^{(k,1)}, \dots, \tilde{x}^{(k,p)}$ or simply by choosing the best of these candidate points or a better point.

Comparison between PVD and PVT reveals that the latter also subsumes the former by letting, for each ℓ ,

$$m_\ell := n_\ell + p - 1,$$

$$A_\ell^{(k)} := \begin{bmatrix} d_1^{(k)} & & & & & \\ & \ddots & & & & \\ & & d_{\ell-1}^{(k)} & & & \\ & & & I_{n_\ell} & & \\ & & & & d_{\ell+1}^{(k)} & \\ & & & & & \ddots \\ & & & & & & d_p^{(k)} \end{bmatrix} \in \mathbb{R}^{n \times (n_\ell+p-1)}.$$

In fact, PVD then corresponds to the resultant PVT with the identification

$$y_\ell = (\xi_1, \dots, \xi_{\ell-1}, x_\ell - x_\ell^{(k)}, \xi_{\ell+1}, \dots, \xi_p), \quad \ell = 1, \dots, p.$$

It is easily seen that condition (A1) holds. Condition (A3) is also satisfied, provided that the sequence $\{d^{(k)}\}$ is bounded.

Note that the original PVD [4] assumes that the subproblems are solved exactly in the parallelization phase. This restrictive requirement has been relaxed by Solodov [8]. Algorithm PVD has also been extended to constrained optimization problems [4, 9].

The UCS method [5] (see also [2]) is an iterative method that minimizes, at each iteration, the quadratic model function

$$(39) \quad q^{(k)}(v) \equiv f(x^{(k)}) + \nabla f(x^{(k)})^\top v + \frac{1}{2} v^\top G^{(k)} v,$$

where $G^{(k)}$ is an $n \times n$ symmetric positive definite matrix. The minimization is carried out by solving the following problems in parallel for $\ell = 1, \dots, p$:

$$(40) \quad \min_{w_\ell \in \mathbb{R}^{n_\ell}} q^{(k)}(Z_\ell^{(k)} w_\ell),$$

where $Z_\ell^{(k)}$ are $n \times n_\ell$ matrices whose columns form bases of subspaces $T_\ell^{(k)}$ of $\mathbb{R}^n$ such that $n_1 + \dots + n_p = n$, $T_1^{(k)} + \dots + T_p^{(k)} = \mathbb{R}^n$, and $x^\top G^{(k)} x' = 0$ for any $x \in T_\ell^{(k)}$, $x' \in T_{\ell'}^{(k)}$ with $\ell \neq \ell'$. The unique minimizers of problems (40) are obtained as

$$(41) \quad w_\ell^{(k)} = - \left((Z_\ell^{(k)})^\top G^{(k)} Z_\ell^{(k)} \right)^{-1} (Z_\ell^{(k)})^\top \nabla f(x^{(k)}), \quad \ell = 1, \dots, p,$$

and the vector

$$(42) \quad v^{(k)} = \sum_{\ell=1}^p Z_\ell^{(k)} w_\ell^{(k)}$$

is precisely the unique minimizer of the quadratic function $q^{(k)}$ defined by (39). In particular, if the Hessian $\nabla^2 f(x^{(k)})$ is positive definite and $G^{(k)}$ is chosen to be $\nabla^2 f(x^{(k)})$, then $v^{(k)}$ is the Newton direction $v^{(k)} = -\nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)})$. With the vector $v^{(k)}$ given by (42), the next iterate is determined as $x^{(k+1)} := x^{(k)} + \alpha v^{(k)}$ for some stepsize $\alpha > 0$.

The PVT algorithm may be specialized to the UCS method by letting, for each ℓ ,

$$\begin{aligned} m_\ell &:= n_\ell, \\ A_\ell^{(k)} &:= Z_\ell^{(k)} \in \mathbb{R}^{n \times n_\ell}. \end{aligned}$$

Moreover, if we define

$$\varphi_\ell^{(k)}(w_\ell) \equiv f(Z_\ell^{(k)} w_\ell + x^{(k)}), \quad \ell = 1, \dots, p,$$

then the positive definiteness of $G^{(k)}$ implies that, for each $\ell = 1, \dots, p$, the vector $w_\ell^{(k)}$ given by (41) satisfies the descent condition

$$\begin{aligned} \nabla \varphi_\ell^{(k)}(0)^\top w_\ell^{(k)} &= \nabla f(x^{(k)})^\top Z_\ell^{(k)} w_\ell^{(k)} \\ &< 0 \end{aligned}$$

whenever $w_\ell^{(k)} \neq 0$. Hence $x^{(k+1)} = x^{(k)} + \alpha v^{(k)} = x^{(k)} + \alpha \sum_{\ell=1}^p Z_\ell^{(k)} w_\ell^{(k)}$ is a particular but plausible way of determining the next iterate under the rule in the synchronization phase of the PVT algorithm. If the columns of each $Z_\ell^{(k)}$ are also mutually conjugate with respect to $G^{(k)}$ and are normalized so that the matrix $Z^{(k)} := [Z_1^{(k)} \dots Z_p^{(k)}] \in \mathbb{R}^{n \times n}$ satisfies $(Z^{(k)})^\top G^{(k)} Z^{(k)} = I$, i.e., $Z^{(k)} (Z^{(k)})^\top = (G^{(k)})^{-1}$, then conditions (A1) and (A3) are satisfied, provided $\{G^{(k)}\}$ is bounded and uniformly positive definite.

6. Specific transformations. In this section, we mention some possibilities of designing new parallel algorithms within the framework of the PVT algorithm. First we give a natural extension of the block Jacobi method. Let J_ℓ , $\ell = 1, \dots, p$, be sets of indices such that

$$(43) \quad J_1 \cup \dots \cup J_p = \{1, \dots, n\}.$$

We note in particular that the sets J_ℓ , $\ell = 1, \dots, p$, are not assumed to be disjoint to each other. In other words, we have in general $J_\ell \cap J_{\ell'} \neq \emptyset$ for some ℓ, ℓ' such that $\ell \neq \ell'$. Let

$$m_\ell := |J_\ell|, \quad \ell = 1, \dots, p,$$

and, for each $\ell \in \{1, \dots, p\}$, let $A_\ell^{(k)}$ be the $n \times m_\ell$ matrix with $\text{rank } A_\ell^{(k)} = m_\ell$ such that, for any $j \in J_\ell$, the j th row of $A_\ell^{(k)}$ is a row of the $m_\ell \times m_\ell$ identity matrix, and the other rows of $A_\ell^{(k)}$ are all zero. For example, if $n = 4$ and $p = 3$ with $J_1 = \{1, 2\}$, $J_2 = \{2, 3\}$, and $J_3 = \{3, 4\}$, then the matrices $A_\ell^{(k)}$, $\ell = 1, 2, 3$, are given by

$$A_1^{(k)} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad A_2^{(k)} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad A_3^{(k)} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

For each ℓ , let $x_\ell \in \mathbb{R}^{m_\ell}$ and $x_{\bar{\ell}} \in \mathbb{R}^{n-m_\ell}$ denote the subvectors of $x \in \mathbb{R}^n$ with those elements that correspond to the index set J_ℓ and its complement in $\{1, \dots, n\}$, respectively. With the identification

$$y_\ell = x_\ell - x_\ell^{(k)}, \quad \ell = 1, \dots, p,$$

the subproblem (2) in the PVT algorithm can then be written as

$$\min_{x_\ell \in \mathbb{R}^{m_\ell}} f(x_\ell, x_\ell^{(k)}),$$

which looks exactly the same as that of the block Jacobi method, except for a possible overlap of variables among subproblems. It is clear that conditions (A1) and (A3) are both satisfied.

The UCS method can also be generalized in a similar manner. Let $G^{(k)}$ be an $n \times n$ symmetric positive definite matrix and $Z^{(k)}$ be an $n \times n$ matrix whose columns are mutually conjugate with respect to $G^{(k)}$. Let $J_\ell, \ell = 1, \dots, p$, be sets of indices satisfying (43), and let $m_\ell := |J_\ell|, \ell = 1, \dots, p$. Again we do not require the index sets J_ℓ to be mutually disjoint. For each $\ell \in \{1, \dots, p\}$, let $Z_\ell^{(k)}$ denote the $n \times m_\ell$ matrix that consists of the columns of $Z^{(k)}$ corresponding to the index set J_ℓ . By regarding $Z_\ell^{(k)}$ as the transformation matrix $A_\ell^{(k)}$ for each ℓ , the subproblems (2) in the PVT algorithm may be written as

$$\min_{y_\ell \in \mathbb{R}^{m_\ell}} f(Z_\ell^{(k)} y_\ell + x^{(k)}), \quad \ell = 1, \dots, p.$$

Note that $A^{(k)} := [Z_1^{(k)} \dots Z_p^{(k)}] \in \mathbb{R}^{n \times (m_1 + \dots + m_p)}$ satisfies the inequalities

$$x^\top Z^{(k)} (Z^{(k)})^\top x \leq x^\top A^{(k)} (A^{(k)})^\top x \leq \gamma x^\top Z^{(k)} (Z^{(k)})^\top x \quad \text{for all } x \in \mathbb{R}^n,$$

where γ is the maximum multiplicity of overlapping columns among $Z_\ell^{(k)}, \ell = 1, \dots, p$. Therefore, just like the original UCS method, if the columns of $Z^{(k)}$ are normalized to satisfy $(Z^{(k)})^\top G^{(k)} Z^{(k)} = I$, i.e., $Z^{(k)} (Z^{(k)})^\top = (G^{(k)})^{-1}$, then conditions (A1) and (A3) hold, provided $\{G^{(k)}\}$ is bounded and uniformly positive definite.

The transformation matrices $A_\ell^{(k)}$ in the generalized block Jacobi method and the generalized UCS method described in this section can be further modified using the idea behind the PVD algorithm. Specifically, we may augment the matrices $A_\ell^{(k)}$ by adding arbitrary columns that are generated using, for example, the steepest descent or quasi-Newton directions in some subspaces of appropriate dimension. As a result, it is expected that each processor obtains a better solution in the parallelization phase at the expense of an increase in the number of variables in subproblem (2).

7. Conclusion. We have presented a general framework for designing parallel algorithms for unconstrained optimization. The proposed PVT algorithm covers in particular the PGD algorithm [6], the unconstrained PVD algorithm [4], and the UCS method [5]. As suggested in section 6, the proposed approach enables us to design an algorithm in an extremely flexible manner. Important future topics include extending the concept of parallel variable transformation to constrained optimization and investigating practical and efficient implementation strategies of the PVT algorithm through extensive computational experiments.

Acknowledgments. I am grateful to the referees for their constructive suggestions and helpful comments.

REFERENCES

- [1] D. P. BERTSEKAS AND J. N. TSITSIKLIS, *Parallel and Distributed Computation: Numerical Methods*, Prentice-Hall, Englewood Cliffs, NJ, 1989.
- [2] M.-Q. CHEN AND S.-P. HAN, *A parallel quasi-Newton method for partially separable large scale minimization*, Ann. Oper. Res., 14 (1988), pp. 195–211.
- [3] A. R. CONN, N. GOULD, A. SARTENAER, AND PH. L. TOINT, *On iterated-subspace minimization methods for nonlinear optimization*, in Linear and Nonlinear Conjugate Gradient-Related Methods, L. M. Adams and J. L. Nazareth, eds., SIAM, Philadelphia, 1996, pp. 50–78.
- [4] M. C. FERRIS AND O. L. MANGASARIAN, *Parallel variable distribution*, SIAM J. Optim., 4 (1994), pp. 102–126.
- [5] S.-P. HAN, *Optimization by updated conjugate subspaces*, in Numerical Analysis: Pitman Research Notes in Mathematics Series 140, D. F. Griffiths and G. A. Watson, eds., Longman Scientific & Technical, Burnt Mill, England, 1986, pp. 82–97.
- [6] O. L. MANGASARIAN, *Parallel gradient distribution in unconstrained optimization*, SIAM J. Control Optim., 33 (1995), pp. 1916–1925.
- [7] J. M. ORTEGA AND W. C. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.
- [8] M. V. SOLODOV, *New inexact parallel variable distribution algorithms*, Comput. Optim. Appl., 7 (1997), pp. 165–182.
- [9] M. V. SOLODOV, *On the convergence of constrained parallel variable distribution algorithms*, SIAM J. Optim., 8 (1998), pp. 187–196.

SOME FEASIBILITY ISSUES IN MATHEMATICAL PROGRAMS WITH EQUILIBRIUM CONSTRAINTS*

MASAO FUKUSHIMA[†] AND JONG-SHI PANG[‡]

Abstract. This paper is concerned with some feasibility issues in mathematical programs with equilibrium constraints (MPECs) where additional joint constraints are present that must be satisfied by the state and design variables of the problems. We introduce sufficient conditions that guarantee the feasibility of these MPECs. It turns out that these conditions also guarantee the feasibility of the quadratic programming (QP) subproblems arising from the penalty interior point algorithm (PIPA) and the sequential quadratic programming (SQP) algorithm for solving MPECs; thus the same conditions ensure that these algorithms are applicable for solving this class of jointly constrained MPECs.

Key words. mathematical programs with equilibrium constraints, penalty interior point algorithm, sequential quadratic programming algorithm

AMS subject classifications. 90C30, 90C33

PII. S105262349731577X

1. Introduction. MPECs have in recent years received growing attention among optimization researchers. The monograph [5] presents a comprehensive study of these important optimization problems and contains an extensive list of references. Up to now, the rigorous study of the MPECs has been to a large extent restricted to the case where the only constraints that link the first-level (design) and second-level (state) variables are the equilibrium constraints; analytical results pertaining to the case where additional joint constraints are present are scarce and incomplete. Our goal in this paper is to examine some basic feasibility issues associated with mathematical programs with affine equilibrium constraints where additional joint linear constraints are present besides the equilibrium constraints.

Specifically, we consider the following MPEC:

$$\begin{aligned}
 (1.1) \quad & \text{minimize} && f(x, y) \\
 & \text{subject to} && Ax + By = b, \\
 & && Nx + My - w = q, \\
 & && w^T y = 0, \\
 & && x \in \mathcal{C}, y \geq 0, w \geq 0,
 \end{aligned}$$

* Received by the editors January 29, 1997; accepted for publication (in revised form) April 11, 1997; published electronically June 3, 1998.

<http://www.siam.org/journals/siopt/8-3/31577.html>

[†] Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan (fuku@kuamp.kyoto-u.ac.jp). This work was based on research supported partly by a Japan-U.S. collaborative grant awarded by the U.S. National Science Foundation under grant INT-9417085 and the Japan Society for the Promotion of Science. The author's research was also supported in part by the Scientific Research Grant-in-Aid from the Ministry of Education, Science, Sports, and Culture of Japan.

[‡] Department of Mathematical Sciences, Whiting School of Engineering, The Johns Hopkins University, Baltimore, MD 21218-2692 (jsp@vicp1.mts.jhu.edu). This work was based on research supported partly by a Japan-U.S. collaborative grant awarded by the U.S. National Science Foundation under grant INT-9417085 and the Japan Society for the Promotion of Science. This author's research was also supported by the U.S. National Science Foundation grant CCR-9624018.

where $f : \mathbb{R}^{n+m} \rightarrow \mathbb{R}$ is a continuously differentiable function, $A \in \mathbb{R}^{p \times n}$, $B \in \mathbb{R}^{p \times m}$, $N \in \mathbb{R}^{m \times n}$, and $M \in \mathbb{R}^{m \times m}$ are given matrices, $b \in \mathbb{R}^p$ and $q \in \mathbb{R}^m$ are given vectors, and $\mathcal{C}$ is a polyhedral convex cone in $\mathbb{R}^n$. In this formulation, x is the first-level (design) variable and y and w are the second-level (state) variables. The equilibrium constraints (also called state constraints) are expressed in the form of the parametric linear complementarity problem (LCP) with x as the parameter and y as the primary variable:

$$(1.2) \quad 0 \leq y \perp w = -q + Nx + My \geq 0,$$

where the notation $a \perp b$ means that the two vectors a and b are perpendicular; for simplicity, the additional joint constraint between the design variable x and the state variable y is assumed to be modeled by the linear equation

$$Ax + By = b.$$

Other joint constraints in the form of inequalities can easily be cast in the above form by introducing slack variables and redefining the first-level variables and the cone $\mathcal{C}$.

The case where $B = 0$ in the problem (1.1) has received a fairly comprehensive treatment in the literature; see [5]. In particular, the implicit programming approach [4, 5], the PIPA [5], a bundle-type algorithm [6], the SQP algorithm [3], and a continuation-type algorithm [2] all deal with this special case. A fundamental reason for this restricted treatment to date lies in the fact that the absence of the matrix B significantly simplifies the feasibility issue of the MPEC itself; algorithmically, this special case relegates the feasibility consideration in the subproblems arising from an iterative method for solving the MPEC such as PIPA or the SQP algorithm. Indeed, the feasibility issue for either problem (1.1) itself or the subproblems in the numerical methods has been the major bottleneck for a complete treatment of the problem (1.1) when B is not the zero matrix.

To better explain this point, consider the MPEC (1.1) and suppose that $B = 0$ and M is a Q-matrix [1]. Then for any $x \in \mathbb{R}^n$, a vector $(y, w) \in \mathbb{R}^{2m}$ satisfying the state constraints (1.2) always exists by the Q-property of M ; hence the MPEC (1.1) is feasible whenever the set $\{x \in \mathcal{C} \mid Ax = b\}$ is nonempty. However, in the general case where $B \neq 0$, little is known to date about conditions that guarantee the feasibility of the MPEC and subproblems of iterative methods for solving it.

In this paper, we introduce conditions under which feasibility of the MPEC (1.1) is guaranteed. We consider the extension of PIPA to the MPEC (1.1) with side joint linear constraints and show that one of the conditions introduced also ensures the feasibility of the QP subproblems to be solved at each iteration of this algorithm. Finally, we do the same for the SQP algorithm.

2. Feasibility of MPEC. The constraint system of the MPEC (1.1) can alternatively be written as

$$(2.1) \quad \begin{aligned} Ax + By &= b, \\ x &\in \mathcal{C}, \\ 0 &\leq y \perp (Nx + My - q) \geq 0. \end{aligned}$$

We introduce two assumptions on this nonlinear constraint system:

(A1) the linear system

$$(2.2) \quad \begin{aligned} Ax + By &= b, \\ x &\in \mathcal{C}, \\ Nx + My - q &\geq 0, \\ y &\geq 0 \end{aligned}$$

is consistent;

(A2) the implication

$$\left. \begin{aligned} A^T u + N^T v &\in \mathcal{C}^* \\ v \circ (B^T u + M^T v) &\leq 0 \end{aligned} \right\} \implies v \circ (B^T u + M^T v) = 0$$

holds, where $\mathcal{C}^*$ denotes the dual cone of $\mathcal{C}$ and $\circ$ denotes the Hadamard product operation.

Since the system (2.2) is a relaxation of (2.1), assumption (A1) is clearly necessary for the constraint system (2.1) to be consistent. The implication in (A2) is reminiscent of the row sufficiency condition of a square matrix in LCP theory [1, section 3.5]. It is interesting to note that it is not necessary for M to be a P_0 -matrix in order for (A2) to hold; a counterexample exists in the case when the matrix $A = 0$, N is the identity matrix, and $\mathcal{C}$ is the entire $\mathbb{R}^n$. In this case, (A2) holds trivially for arbitrary matrices B and M .

The following two propositions provide some interesting sufficient conditions for assumption (A2) to hold.

PROPOSITION 2.1. *Let the matrix*

$$\begin{pmatrix} 0 & B \\ B^T & 2M \end{pmatrix}$$

be copositive on the cone $\{(u, v) \in \mathbb{R}^{p+m} \mid A^T u + N^T v \in \mathcal{C}^*\}$, i.e.,

$$(2.3) \quad A^T u + N^T v \in \mathcal{C}^* \implies (u^T, v^T) \begin{pmatrix} 0 & B \\ B^T & 2M \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} \geq 0.$$

Then assumption (A2) is satisfied.

Proof. If (u, v) satisfies

$$A^T u + N^T v \in \mathcal{C}^*,$$

then (2.3) implies

$$v^T (B^T u + M^T v) \geq 0.$$

So if (u, v) also satisfies

$$v \circ (B^T u + M^T v) \leq 0,$$

we must have

$$v \circ (B^T u + M^T v) = 0.$$

The proof is complete. $\square$

PROPOSITION 2.2. *Assume that M is positive semidefinite and that for any vector $u \in \mathbb{R}^p$, there exists a vector $\hat{x} \in \mathbb{R}^n$ such that*

$$(2.4) \quad A\hat{x} = 0, \quad N\hat{x} = B^T u, \quad \hat{x} \in \mathcal{C}.$$

Then assumption (A2) is satisfied.

Proof. Let (u, v) satisfy

$$\begin{aligned} A^T u + N^T v &\in \mathcal{C}^*, \\ v \circ (B^T u + M^T v) &\leq 0. \end{aligned}$$

The last inequality implies

$$v^T B^T u + v^T M^T v \leq 0.$$

Therefore, it follows from (2.4) that

$$0 \leq u^T A\hat{x} + v^T N\hat{x} = v^T B^T u \leq -v^T M^T v \leq 0,$$

where the last inequality follows from the positive semidefiniteness of M . Thus we obtain

$$v^T M^T v = v^T B^T u = 0,$$

which implies that assumption (A2) is satisfied. $\square$

REMARK. Geometrically, the expression (2.4) stipulates that the column space of matrix B^T is contained in the image of the set $\mathcal{C} \cap \ker(A)$ under the linear transformation defined by matrix N . Thus the larger the range of the matrix N is or the smaller the range of the matrix B^T is, the more likely it is that this condition can be satisfied.

Next we establish the main theorem that gives a sufficient condition for the MPEC (1.1) to be feasible.

THEOREM 2.1. *Under assumptions (A1) and (A2), the constraint system (2.1) is consistent.*

Proof. It suffices to show that the following quadratic program in the variable $(x, y) \in \mathbb{R}^{n+m}$ has an optimal solution with zero objective value:

$$\begin{aligned} &\text{minimize} && y^T (Nx + My - q) \\ &\text{subject to} && (2.2). \end{aligned}$$

By assumption (A1) and the well-known Frank–Wolfe theorem in QP, this program has an optimal solution, say, $(\bar{x}, \bar{y})$, which together with some multiplier vector $(\lambda, \mu) \in \mathbb{R}^{p+m}$ satisfies the complementarity system:

$$\begin{aligned} 0 &\leq \bar{y} \perp [N\bar{x} + M\bar{y} - q + B^T \lambda + M^T(\bar{y} - \mu)] \geq 0, \\ \mathcal{C} &\ni \bar{x} \perp [A^T \lambda + N^T(\bar{y} - \mu)] \in \mathcal{C}^*, \\ 0 &\leq \mu \perp [N\bar{x} + M\bar{y} - q] \geq 0. \end{aligned}$$

To simplify the notation, we denote

$$\begin{aligned} \phi &:= N\bar{x} + M\bar{y} - q, \\ \psi &:= B^T \lambda + M^T(\bar{y} - \mu). \end{aligned}$$

Then the above complementarity system can be written as

$$(2.5) \quad 0 \leq \bar{y} \perp [\phi + \psi] \geq 0,$$

$$(2.6) \quad \mathcal{C} \ni \bar{x} \perp [A^T \lambda + N^T(\bar{y} - \mu)] \in \mathcal{C}^*,$$

$$(2.7) \quad 0 \leq \mu \perp \phi \geq 0.$$

From (2.5) and (2.7), we have

$$(2.8) \quad \bar{y} \circ \phi \geq 0,$$

$$(2.9) \quad \bar{y} \circ [\phi + \psi] = 0,$$

$$(2.10) \quad \mu \circ \phi = 0,$$

$$(2.11) \quad \mu \circ [\phi + \psi] \geq 0.$$

Thus we can deduce

$$(2.12) \quad (\bar{y} - \mu) \circ \psi \leq (\bar{y} - \mu) \circ [\phi + \psi] \leq 0,$$

where the first inequality follows from (2.8) and (2.10) and the second inequality follows from (2.9) and (2.11). By assumption (A2), it follows from (2.6) and (2.12) that

$$(2.13) \quad (\bar{y} - \mu)^T \psi = 0.$$

Therefore, we get

$$\bar{y}^T \phi = -\bar{y}^T \psi = -\mu^T \psi = -\mu^T (\phi + \psi),$$

where the equalities follow from (2.9), (2.13), and (2.10), respectively. Consequently, by (2.8) and (2.11), we have $\bar{y}^T \phi = 0$, as desired. $\square$

3. Feasibility of PIPA subproblems. In this section, we study conditions for all the PIPA QP subproblems to be feasible, therefore, for the applicability of the PIPA for solving the MPEC (1.1) with joint linear constraints on the state and design variables. We refer the reader to [5, Chapter 6] for the detailed description of PIPA. The important point to be made here is that in this reference, the analysis of PIPA is restricted to the case where $B = 0$ and M is a P_0 -matrix. The purpose of this section is to extend this analysis by removing both of these restrictions.

It will be useful for us to first summarize the essentials of PIPA for solving the MPEC (1.1). This algorithm is designed to trace the solution path

$$\{(x(\mu), y(\mu), w(\mu)) \in \mathbb{R}^{n+2m} \mid \mu > 0\}$$

of the parametrized family of nonlinear programs in which the complementarity conditions $w^T y = 0$, $y \geq 0$, $w \geq 0$ are replaced by $w \circ y = 2\mu e$, $y > 0$, $w > 0$, where μ is a positive parameter approaching zero and $e \in \mathbb{R}^m$ is the vector of all ones. Specifically, the vector $(x(\mu), y(\mu), w(\mu))$ is defined as a solution of the nonlinear program:

$$(3.1) \quad \begin{aligned} & \text{minimize} && f(x, y) \\ & \text{subject to} && Ax + By = b, \\ & && Nx + My - w = q, \\ & && w \circ y = 2\mu e, \\ & && x \in \mathcal{C}, y > 0, w > 0. \end{aligned}$$

In practice, since it is not easy to trace this parametric solution curve, a sequence of approximate iterates $\{(x^k, y^k, w^k)\}$ is generated as follows. Let $(x^k, y^k, w^k) \in \mathfrak{R}^{n+2m}$ be the current iterate satisfying the following conditions

$$(3.2) \quad Ax^k + By^k = b, \quad Nx^k + My^k - w^k = q, \quad x^k \in \mathcal{C}, \quad y^k > 0, \quad w^k > 0.$$

Let Q_k be an $(n+2m) \times (n+2m)$ symmetric positive definite matrix. Then at each iteration of PIPA we compute the search direction $(dx^k, dy^k, dw^k) \in \mathfrak{R}^{n+2m}$ by solving the QP subproblem

$$(3.3) \quad \begin{aligned} & \text{minimize} \quad \nabla f(x^k, y^k)^T \begin{pmatrix} dx \\ dy \\ dw \end{pmatrix} + \frac{1}{2} (dx^T, dy^T, dw^T) Q_k \begin{pmatrix} dx \\ dy \\ dw \end{pmatrix} \\ & \text{subject to} \quad Adx + Bdy = 0, \\ & \quad \quad \quad Ndx + Mdy - dw = 0, \\ & \quad \quad \quad W^k dy + Y^k dw = -Y^k w^k + \sigma_k \mu_k e, \\ & \quad \quad \quad x^k + dx \in \mathcal{C}, \end{aligned}$$

where W^k and Y^k denote the diagonal matrices with diagonal entries w_i^k and y_i^k , respectively, $\sigma_k \in (0, 1)$ is the centering step size that controls the direction (dx^k, dy^k, dw^k) between the pure Newton direction ($\sigma_k = 0$) and the perfectly central direction ($\sigma_k = 1$), and μ_k is the positive scalar given by $\mu_k := (w^k)^T y^k / m$.

In the following, we study conditions under which the QP subproblem (3.3) has a feasible solution for each k . The positive definiteness of the matrix Q_k then ensures that (3.3) has a unique optimal solution. Upon substituting $z := x^k + dx$ and eliminating dw , the constraints of the QP subproblem (3.3) can be rewritten as

$$(3.4) \quad \begin{aligned} Az + Bdy &= Ax^k, \\ Nz + (M + (Y^k)^{-1}W^k)dy &= s^k, \\ z &\in \mathcal{C}, \end{aligned}$$

where s^k is given by

$$s^k := Nx^k - w^k + \sigma_k \mu_k (Y^k)^{-1} e.$$

The goal of the following analysis is to answer the question as to what conditions will ensure that the system (3.4) is consistent for an arbitrary s^k . The following lemma plays an important role in answering this question.

LEMMA 3.1. *The system (3.4) is consistent for any $s^k \in \mathfrak{R}^m$ if and only if*

$$(3.5) \quad \left. \begin{aligned} & A^T u + N^T v \in \mathcal{C}^* \\ & B^T u + (M + (Y^k)^{-1}W^k)^T v = 0 \end{aligned} \right\} \implies v = 0.$$

Proof. By the generalized Farkas lemma, the system (3.4) is consistent if and only if

$$\left. \begin{aligned} & A^T u + N^T v \in \mathcal{C}^* \\ & B^T u + (M + (Y^k)^{-1}W^k)^T v = 0 \end{aligned} \right\} \implies (Ax^k)^T u + (s^k)^T v \geq 0.$$

Then it is easy to observe that this implication holds for any $s^k \in \mathfrak{R}^m$ if and only if

$$(3.6) \quad \left. \begin{array}{l} A^T u + N^T v \in \mathcal{C}^* \\ B^T u + (M + (Y^k)^{-1} W^k)^T v = 0 \end{array} \right\} \implies (Ax^k)^T u \geq 0, \quad v = 0.$$

We show that (3.6) is equivalent to the implication (3.5). It is trivial that (3.6) implies (3.5). Conversely, suppose that (3.5) holds. If (u, v) satisfies the left-hand side of the implication (3.6), then it follows from (3.5) that $v = 0$, which in turn implies that $A^T u \in \mathcal{C}^*$. Since $x^k \in \mathcal{C}$ by (3.2), we obtain $(Ax^k)^T u = (x^k)^T A^T u \geq 0$. Hence (3.6) holds. $\square$

Now we show that assumption (A2) given in the previous section also provides a sufficient condition for the QP subproblem (3.3) to be feasible for all k .

THEOREM 3.1. *Under assumption (A2), the QP subproblem (3.3) is feasible for all k .*

Proof. According to Lemma 3.1, it suffices to show that implication (3.5) holds for each k . Let (u, v) satisfy the left-hand side of implication (3.5). Then since

$$B^T u + M^T v = -(Y^k)^{-1} W^k v,$$

we have

$$(3.7) \quad v \circ [B^T u + M^T v] = -v \circ [(Y^k)^{-1} W^k v] \leq 0,$$

where the inequality holds because Y^k and W^k are positive diagonal. Assumption (A2) then yields

$$v^T (B^T u + M^T v) = 0,$$

which together with (3.7) implies

$$v^T (Y^k)^{-1} W^k v = 0.$$

Hence we have $v = 0$. This completes the proof. $\square$

4. Feasibility of SQP subproblems. The results obtained in the previous section extend to the SQP algorithm proposed in [3]. In the SQP algorithm, we consider another parametrized family of nonlinear programs:

$$(4.1) \quad \begin{array}{ll} \text{minimize} & f(x, y) \\ \text{subject to} & Ax + By = b, \\ & Nx + My - w = q, \\ & \Phi(y, w, \mu) = 0, \\ & x \in \mathcal{C}, \end{array}$$

where μ is a nonnegative parameter and the function $\Phi : \mathfrak{R}^{2m} \times [0, \infty) \rightarrow \mathfrak{R}^m$ is defined by

$$\Phi(y, w, \mu) := \begin{pmatrix} y_1 + w_1 - (y_1^2 + w_1^2 + \mu)^{1/2} \\ \vdots \\ y_m + w_m - (y_m^2 + w_m^2 + \mu)^{1/2} \end{pmatrix}.$$

When $\mu = 0$, problem (4.1) is equivalent to the original MPEC (1.1). However, since the function $\Phi(\cdot, \cdot, 0)$ is not differentiable everywhere, it is not easy to solve the nonlinear program (4.1) with $\mu = 0$ directly. Instead, the SQP algorithm solves a sequence of QP subproblems, each of which is an approximation to problem (4.1) with some μ strictly positive but approaching zero. Specifically, let μ_k be a positive scalar, (x^k, y^k, w^k) be the current iterate satisfying

$$(4.2) \quad Ax^k + By^k = b, \quad Nx^k + My^k - w^k = q, \quad x^k \in \mathcal{C}$$

(unlike PIPA, the vector (y^k, w^k) is not required to be positive), and Q_k be an $(n + 2m) \times (n + 2m)$ symmetric positive definite matrix. Then the QP subproblem for the SQP algorithm can be written as

$$(4.3) \quad \begin{aligned} & \text{minimize} \quad \nabla f(x^k, y^k)^T \begin{pmatrix} dx \\ dy \end{pmatrix} + \frac{1}{2} (dx^T, dy^T, dw^T) Q_k \begin{pmatrix} dx \\ dy \\ dw \end{pmatrix} \\ & \text{subject to} \quad Adx + Bdy = 0, \\ & \quad \quad \quad Ndx + Mdy - dw = 0, \\ & \quad \quad \quad D_a^k dy + D_b^k dw = -\Phi(y^k, w^k, \mu_k), \\ & \quad \quad \quad x^k + dx \in \mathcal{C}, \end{aligned}$$

where D_a^k and D_b^k denote the diagonal matrices with diagonal entries

$$1 - \frac{y_i^k}{\sqrt{(y_i^k)^2 + (w_i^k)^2 + \mu_k}} \quad \text{and} \quad 1 - \frac{w_i^k}{\sqrt{(w_i^k)^2 + (w_i^k)^2 + \mu_k}},$$

respectively. Since these entries are clearly positive, problem (4.3) has the same form as problem (3.3). Hence exactly the same analysis for PIPA yields the following theorem concerning the feasibility of the QP subproblem (4.3) in the SQP algorithm. The proof is omitted.

THEOREM 4.1. *Under assumption (A2), the QP subproblem (4.3) is feasible for all k .*

5. Concluding remarks. We conclude this paper with some comments that arise from Theorems 3.1 and 4.1. These results have identified a class of MPECs with side joint constraints to which PIPA and SQP algorithm are applicable (again, see [5] and [3] for details of these algorithms). More importantly, this class of MPECs does not require M to be a P_0 -matrix (see the discussion preceding Proposition 2.1). This generality raises the question of whether the convergence proofs for PIPA and SQP algorithm that require the P_0 -property of M as given in [5, Chapter 6] and [3], respectively, will remain valid for the more general MPEC (1.1) with assumption (A2) replacing the P_0 -assumption. This issue is presently being investigated.

Although it is not clear to us at this time how easily satisfied assumption (A2) is in practical applications of the MPEC (1.1), this condition is nevertheless the first of its kind in the study of the general MPEC. We are hopeful that further results of this nature can be obtained and that our work will provide a lead in this direction of research for the MPEC.

REFERENCES

- [1] R. W. COTTLE, J.-S. PANG, AND R. E. STONE, *The Linear Complementarity Problem*, Academic Press, Boston, MA, 1992.
- [2] F. FACCHINEI, H. JIANG, AND L. QI, *A smoothing method for mathematical programs with equilibrium constraints*, Math. Programming, to appear.
- [3] M. FUKUSHIMA, Z.-Q. LUO, AND J.-S. PANG, *A globally convergent sequential programming algorithm for mathematical programs with linear complementarity constraints*, Comput. Optim. Appl., 10 (1998), pp. 5–34.
- [4] J. S. PANG, S. P. HAN, AND N. RANGARAJ, *Minimization of locally Lipschitzian functions*, SIAM J. Optim., 1 (1991), pp. 57–82.
- [5] Z.-Q. LUO, J.-S. PANG, AND D. RALPH, *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, New York, 1996.
- [6] J. V. OUTRATA AND J. ZOWE, *A numerical approach to optimization problems with variational inequality constraints*, Math. Programming, 68 (1995), pp. 105–130.

ON THE IMPLEMENTATION OF AN ALGORITHM FOR LARGE-SCALE EQUALITY CONSTRAINED OPTIMIZATION*

MARUCHA LALEE[†], JORGE NOCEDAL[†], AND TODD PLANTENGA[‡]

Abstract. This paper describes a software implementation of Byrd and Omojokun's trust region algorithm for solving nonlinear equality constrained optimization problems. The code is designed for the efficient solution of large problems and provides the user with a variety of linear algebra techniques for solving the subproblems occurring in the algorithm. Second derivative information can be used, but when it is not available, limited memory quasi-Newton approximations are made. The performance of the code is studied using a set of difficult test problems from the CUTE collection.

Key words. minimization, nonlinear optimization, large-scale optimization, constrained optimization, trust region methods, quasi-Newton methods

AMS subject classifications. 65K05, 90C30

PII. S1052623493262993

1. Introduction. This paper describes the implementation and testing of an algorithm for solving large nonlinear optimization problems with equality constraints. We write this problem as

$$(1.1) \quad \min_{x \in \mathbf{R}^n} f(x) \quad \text{subject to} \quad c(x) = 0,$$

where $f : \mathbf{R}^n \rightarrow \mathbf{R}$ and $c : \mathbf{R}^n \rightarrow \mathbf{R}^m$ are twice continuously differentiable functions, and where $n \geq m$. The algorithm is an extension of the trust region method proposed by Byrd [2] and Omojokun [32], and our implementation is designed for large-scale applications. By this we mean problems where the matrices of function derivatives are more profitably treated using sparse data structures. We have in mind problems with a thousand or more unknowns, and the number of constraints may be small, large, or even equal to n —the algorithm treats all cases in a uniform manner.

Every iteration of the algorithm of Byrd and Omojokun requires the solution of two trust region subproblems of smaller dimension. We provide both iterative and direct solvers for these subproblems, allowing the user to select the most efficient solver for a particular problem. If second derivatives of f and c are not provided, our code automatically generates limited memory BFGS (ℓ -BFGS) approximations.

Inequality constraints are not considered in this study for two reasons. First, we want to determine if the method of Byrd and Omojokun is robust, and we wish to give careful attention to its proper implementation in the setting of large-scale optimization, starting with the simpler case of equality constraints. As we will discuss below, this case raises many difficult questions and requires a sophisticated implementation. The second reason is that the algorithm developed in this paper forms the core of two new algorithms for handling general inequality constraints that are currently under investigation [34], [3].

*Received by the editors February 15, 1994; accepted for publication (in revised form) March 3, 1997; published electronically June 3, 1998. This work was supported by National Science Foundation grants CCR-9101359 and ASC-9213149 and by Department of Energy grant DE-FG02-87ER25047.

<http://www.siam.org/journals/siopt/8-3/26299.html>

[†]Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL 60208 (nocedal@ece.nwu.edu).

[‡]Sandia National Laboratories, MS 9214, P.O. Box 969, Livermore, CA 94551-0969 (tdplant@ca.sandia.gov).

The algorithm is motivated and described in section 2. Two trust region subproblems arise in the algorithm, and section 3 digresses to develop some techniques for finding approximate solutions for them. In section 4 the details of our implementation are presented. The results of numerical experiments, which suggest that the method holds great promise, are discussed in section 5. A summarizing discussion and ideas for further enhancements presented in section 6 conclude the paper.

2. General description of the algorithm. We begin by introducing some notation. The gradient of the objective function f is denoted by g , and A denotes the $n \times m$ matrix of constraint gradients, i.e.,

$$A(x) = [\nabla c^1(x), \dots, \nabla c^m(x)],$$

where c^i , $i = 1, \dots, m$ are the components of the vector c . We will assume that $A(x_k)$ has full column rank for all x_k . The Lagrangian of problem (1.1) is $L(x, \lambda) = f(x) - \lambda^T c(x)$, where λ is the vector of Lagrange multipliers. Throughout the paper $\|\cdot\|$ denotes the ℓ_2 norm.

The algorithm of Byrd [2] and Omojokun [32] can be interpreted as a sequential quadratic programming method with a trust region, and it is derived from earlier work by Vardi [42]; Celis, Dennis, and Tapia [7]; Powell and Yuan [38], and Byrd, Schnabel, and Schultz [5]. (Although motivated differently, [45] and [12] present related methods.) The algorithm decomposes each constrained SQP subproblem into two smaller unconstrained trust region subproblems which are easier to solve. This makes the Byrd–Omojokun method attractive for large-scale optimization, which was one of the reasons why we chose to base our implementation on it.

The algorithm is simple to explain and motivate. At a current iterate x_k we choose a trust region radius Δ_k and Lagrange multipliers λ_k , and attempt to generate a new step d_k by solving

$$(2.1) \quad \min_{d \in \mathbf{R}^n} \quad d^T g_k + \frac{1}{2} d^T \nabla_x^2 L(x_k, \lambda_k) d$$

$$(2.2) \quad \text{subject to} \quad A_k^T d + c_k = 0,$$

$$(2.3) \quad \|d\| \leq \Delta_k,$$

where the subscript denotes evaluation of a function at x_k . However, as is well known, restricting the size of the step by (2.3) may preclude us from satisfying the linear constraint (2.2). Therefore Byrd and Omojokun first compute a step that lies well within the trust region and that satisfies the linear constraint (2.2) as much as possible. This is done by defining a relaxation parameter $\zeta \in (0, 1)$ and computing a step v_k that solves the *vertical* (or *normal*) *subproblem*

$$(2.4) \quad \min_{v \in \mathbf{R}^n} \quad \|A_k^T v + c_k\|$$

$$(2.5) \quad \text{subject to} \quad \|v\| \leq \zeta \Delta_k.$$

This problem can have many solutions, but we will show in section 4.1 that it always has a solution v_k in the range space of A_k ; i.e., v_k will be expressed as a linear combination of the columns of A_k . This allows us to completely decouple this subproblem from the next one.

The algorithm is designed so that the full step d need not move any closer to the feasible manifold than v_k does, so we next reformulate (2.1)–(2.3) as

$$(2.6) \quad \min_{d \in \mathbf{R}^n} \quad d^T g_k + \frac{1}{2} d^T \nabla_x^2 L(x_k, \lambda_k) d$$

$$(2.7) \quad \text{subject to} \quad A_k^T d = A_k^T v_k,$$

$$(2.8) \quad \|d\| \leq \Delta_k.$$

This problem, unlike (2.1)–(2.3), always has a nonempty feasible region (for instance, $d = v_k$ satisfies all constraints). Byrd and Omojokun solve for d by seeking a step complementary to v_k . To this end we compute an $n \times (n - m)$ matrix Z_k that spans the null space of A_k (so that $A_k^T Z_k = 0$), and we define the total step of the algorithm as $d = v_k + Z_k u$, where the vector $u \in \mathbf{R}^{n-m}$ is yet to be determined. Substituting for d in (2.6)–(2.8), noting that v_k and $Z_k u$ are orthogonal, and ignoring constant terms, we obtain

$$(2.9) \quad \min_{u \in \mathbf{R}^{n-m}} \quad (g_k + \nabla_x^2 L_k v_k)^T Z_k u + \frac{1}{2} u^T Z_k^T \nabla_x^2 L_k Z_k u$$

$$(2.10) \quad \text{subject to} \quad \|Z_k u\| \leq \sqrt{\Delta_k^2 - \|v_k\|^2}.$$

We denote the solution of this *horizontal* (or *tangential*) *subproblem* by u_k , and we define the total step as

$$(2.11) \quad d_k = v_k + Z_k u_k.$$

We then set

$$x_{k+1} = x_k + d_k,$$

provided x_{k+1} gives a reduction in the merit function; otherwise, the trust region is reduced and a new trial step is computed. (The merit function will be described later in section 4.6.)

The approach of Byrd–Omojokun thus consists of replacing (2.1)–(2.3) with two trust region subproblems of smaller dimension, each with just a single quadratic constraint. The vertical subproblem (2.4)–(2.5) has a spherical trust region in $\mathbf{R}^n$, whereas the horizontal subproblem (2.9)–(2.10) has an ellipsoidal trust region in $\mathbf{R}^{n-m}$. Both subproblems appear to be much easier to solve than the trust region formulations studied by Celis, Dennis, and Tapia [7], and by Powell and Yuan [38], which include an additional quadratic constraint.

Let us compare the Byrd–Omojokun algorithm with some other methods for large-scale constrained optimization. The trust region method implemented in LANCELOT [9] is based on the minimization of an augmented Lagrangian and is quite different from the SQP approach. The SL1QP method developed by Fletcher [18] also poses a single unconstrained trust region subproblem of dimension n , but it differs markedly from our approach in that it requires the minimization of a nondifferentiable model. SNOPT [23], a new line search implementation of the sequential quadratic programming method, appears to hold much promise, but the computational issues for large-scale problems are fundamentally different from those arising in our trust region method.

The second reason for our interest in the Byrd–Omojokun method (the first reason is the simplicity of the trust region subproblems) is its ability to handle nonlinear constraints. Note that when $n = m$, i.e., when the problem reduces to that of finding the root of a system of nonlinear equations, the Byrd–Omojokun algorithm coincides

with the Levenberg–Marquardt method, which is known to be robust and efficient. Most of the algorithms mentioned in the previous paragraph do not have this property, and we believe that our code will prove to be very effective at handling highly nonlinear constraints.

An outline of the algorithm studied in this paper is given below (section 4.9 contains the detailed version).

ALGORITHM 2.1. *General description of the Byrd–Omojokun algorithm*

Constants $\epsilon > 0$ and $\eta \in (0, 1)$ are given

Choose x_0 and $\Delta_0 > 0$

loop, starting with $k = 0$

 Compute f_k, c_k, g_k, A_k , and Z_k

 Compute multipliers λ_k

if $\|g_k - A_k \lambda_k\|_\infty < \epsilon$ **and** $\|c_k\|_\infty < \epsilon$ **then stop**

 Compute v_k by solving the vertical subproblem (2.4)–(2.5) approximately

 Compute $\nabla_x^2 L_k(x_k, \lambda_k)$ or update its ℓ -BFGS approximation

 Compute u_k by solving the horizontal subproblem (2.9)–(2.10) approximately

 Set $d_k = v_k + Z_k u_k$

 Compute the actual reduction in the merit function, a_red,

 and the predicted reduction, p_red

if $\frac{\text{a\_red}}{\text{p\_red}} \geq \eta$

then set $x_{k+1} = x_k + d_k$, $\Delta_{k+1} \geq \Delta_k$

else set $x_{k+1} = x_k$, $\Delta_{k+1} < \|d_k\|$

continue loop, after incrementing k

We show in the rest of the paper that this algorithm can be converted into efficient and robust software for large-scale computations. Since the key to good performance lies in finding approximate solutions to the subproblems without incurring a high computational cost, we begin by studying a formulation of the trust region problem that is general enough for our purposes.

3. Solving trust region subproblems. We wish to develop efficient methods for approximately solving the trust region problem

$$(3.1) \quad \min_{s \in \mathbf{R}^q} \phi(s) = g^T s + \frac{1}{2} s^T B s \quad \text{subject to} \quad \|Cs\| \leq \Delta,$$

where B is symmetric, C is a $p \times q$ matrix of full rank with $p \geq q$, and where the number of variables q is assumed to be large. The vertical subproblem (2.4)–(2.5) can be written in this form with $p = q = n$ and $C = I$, and the horizontal subproblem (2.9)–(2.10) can be obtained by setting $C = Z_k$, $q = n - m$, and $p = n$.

The presence of C can make the trust region ellipsoidal, and it is therefore convenient to transform the problem so that it has a spherical trust region. Let us define

$$(3.2) \quad \tilde{s} \equiv (C^T C)^{1/2} s,$$

so that

$$(3.3) \quad s = (C^T C)^{-1/2} \tilde{s}.$$

Using (3.3) in (3.1) we obtain the equivalent problem

$$(3.4) \quad \min_{\tilde{s} \in \mathbf{R}^q} \tilde{\phi}(\tilde{s}) = \tilde{g}^T \tilde{s} + \frac{1}{2} \tilde{s}^T \tilde{B} \tilde{s} \quad \text{subject to} \quad \|\tilde{s}\| \leq \Delta,$$

where we have defined

$$(3.5) \quad \tilde{g} \equiv (C^T C)^{-1/2} g \quad \text{and} \quad \tilde{B} \equiv (C^T C)^{-1/2} B (C^T C)^{-1/2}.$$

We will now state some well-known methods for approximately solving the spherically constrained trust region problem (3.4) and then transform them via (3.2) to obtain methods for solving the ellipsoidally constrained problem (3.1). We begin by discussing the Cauchy step, which plays a fundamental role in all of these methods.

Cauchy point. The Cauchy point for (3.4) is defined as the minimizer of $\tilde{\phi}$ in the direction of steepest descent at $\tilde{s} = 0$, subject to the trust constraint. Therefore, we solve the problem

$$\min_{\alpha > 0} -\tilde{g}^T(\alpha \tilde{g}) + \frac{1}{2}(\alpha \tilde{g})^T \tilde{B}(\alpha \tilde{g}) \quad \text{subject to} \quad \|\alpha \tilde{g}\| \leq \Delta$$

to obtain

$$\tilde{s}_{cp} = -\tilde{\alpha} \tilde{g},$$

where

$$(3.6) \quad \tilde{\alpha} = \begin{cases} \frac{\tilde{g}^T \tilde{g}}{\tilde{g}^T \tilde{B} \tilde{g}} & \text{if } \tilde{g}^T \tilde{B} \tilde{g} > 0 \quad \text{and} \quad \frac{(\tilde{g}^T \tilde{g})^{3/2}}{\tilde{g}^T \tilde{B} \tilde{g}} \leq \Delta, \\ \frac{\Delta}{\|\tilde{g}\|} & \text{otherwise.} \end{cases}$$

(If $\|\tilde{g}\| = 0$ then we take $\tilde{s}_{cp} = 0$.) We now apply the transformations (3.2) and (3.5) to obtain the Cauchy step for the ellipsoidal problem (3.1),

$$(3.7) \quad s_{cp} = -\tilde{\alpha} (C^T C)^{-1} g,$$

where $\tilde{\alpha}$ is defined in terms of g and B by (3.6) and (3.5). Thus the transformation results in a Cauchy step in the direction of scaled steepest descent, the scale factor $(C^T C)^{-1}$ coming from the ellipsoidal shape of the trust region constraint.

We next describe the two methods used in our code for approximately solving the trust region problem (3.1) when the number of unknowns q is large: Powell's dogleg method and Steihaug's conjugate gradient (CG) iteration.

Dogleg method. If the Hessian matrix B in (3.1) is positive definite, then Powell's dogleg method [35] can give an inexpensive approximate solution. The method calculates the Cauchy point s_{cp} and the Newton step $s_n = -B^{-1}g$ (even if s_n violates the trust region constraint). The dogleg path consists of the two line segments from $s = 0$ to $s = s_{cp}$ and from $s = s_{cp}$ to $s = s_n$. The dogleg method finds the minimizer of ϕ along this path subject to $\|s\| \leq \Delta$. Since ϕ decreases monotonically along the path, we simply find the intersection point with the trust region boundary, or we use the Newton step if the path lies entirely inside the trust region [13, section 6.4.2]. It is well known that, in the unconstrained setting, the dogleg method is sufficiently accurate to ensure first-order global convergence and a quadratic convergence rate [39]; the same properties carry over to the Byrd-Omojokun algorithm [33]. The cost of computing the Newton step can be high in large problems, and an approximation to it is an attractive option.

Steihaug's method. If the matrix B is not positive definite, there is no unconstrained minimizer of ϕ , and the Newton step s_n is not defined. In this case one can use an extension of CG proposed by Steihaug [40] (some elements of this method were also suggested by Toint [41]). The method is shown for the spherically constrained problem (3.4) in Algorithm 3.1.

ALGORITHM 3.1. *Steihaug's method for problem (3.4)*

Constant $\tilde{\epsilon} > 0$ is given

Start with $\tilde{s}_0 = 0$, $\tilde{r}_0 = -\tilde{g}$, and $\tilde{p}_0 = \tilde{r}_0$

if $\|\tilde{r}_0\| < \tilde{\epsilon}$ **then return** $\tilde{s}_0$

loop, starting with $j = 0$

if $\tilde{p}_j^T \tilde{B} \tilde{p}_j \leq 0$

then find $\tilde{\tau}$ so that $\tilde{s} = \tilde{s}_j + \tilde{\tau} \tilde{p}_j$ minimizes $\tilde{\phi}(\tilde{s})$
and satisfies $\|\tilde{s}\| = \Delta$, and **return** $\tilde{s}$

$\tilde{\alpha}_j = \tilde{r}_j^T \tilde{r}_j / \tilde{p}_j^T \tilde{B} \tilde{p}_j$

$\tilde{s}_{j+1} = \tilde{s}_j + \tilde{\alpha}_j \tilde{p}_j$

if $\|\tilde{s}_{j+1}\| \geq \Delta$

then find $\tilde{\tau} \geq 0$ so that $\|\tilde{s}_j + \tilde{\tau} \tilde{p}_j\| = \Delta$,
and **return** $\tilde{s}_j + \tilde{\tau} \tilde{p}_j$

$\tilde{r}_{j+1} = \tilde{r}_j - \tilde{\alpha}_j \tilde{B} \tilde{p}_j$

if $\|\tilde{r}_{j+1}\| / \|\tilde{r}_0\| < \tilde{\epsilon}$ **then return** $\tilde{s}_{j+1}$

$\tilde{\beta}_{j+1} = \tilde{r}_{j+1}^T \tilde{r}_{j+1} / \tilde{r}_j^T \tilde{r}_j$

$\tilde{p}_{j+1} = \tilde{r}_{j+1} + \tilde{\beta}_{j+1} \tilde{p}_j$

continue loop, after incrementing j

Note that when the trust region constraint is inactive and $\tilde{B}$ is positive definite, Algorithm 3.1 reduces to a CG computation of the Newton step. Steihaug proved [40] that $\tilde{\phi}$ decreases monotonically with each step $\tilde{s}_j$, and that every step moves farther away from the start point $\tilde{s}_0 = 0$, in the sense that $\|\tilde{s}_{j+1}\| > \|\tilde{s}_j\|$. These two properties imply that stopping the iteration as soon as the trust region is encountered is a sensible strategy. Observe that the first step $\tilde{s}_1$ is simply the Cauchy step $\tilde{s}_{cp}$ (assuming $\|\tilde{g}\| \geq \tilde{\epsilon}$), and that the residual vector $\tilde{r}_j$ equals the gradient of $\tilde{\phi}$ at $\tilde{s}_j$.

To adapt Steihaug's method for ellipsoidally constrained problem (3.1), we apply the transformations (3.2) and (3.5). If we additionally define the vectors

$$r_j \equiv (C^T C)^{1/2} \tilde{r}_j \quad \text{and} \quad p_j \equiv (C^T C)^{-1/2} \tilde{p}_j,$$

then we obtain Algorithm 3.2.

ALGORITHM 3.2. *Steihaug's method for problem (3.1)*

Constant $\epsilon > 0$ is given

Start with $s_0 = 0$, $r_0 = -g$, and $p_0 = (C^T C)^{-1} r_0$

if $\sqrt{r_0^T (C^T C)^{-1} r_0} < \epsilon$ **then return** s_0

loop, starting with $j = 0$

if $p_j^T B p_j \leq 0$

then find τ so that $s = s_j + \tau p_j$ minimizes $\phi(s)$
and satisfies $\|Cs\| = \Delta$, and **return** s

$\alpha_j = r_j^T (C^T C)^{-1} r_j / p_j^T B p_j$

$s_{j+1} = s_j + \alpha_j p_j$

if $\|Cs_{j+1}\| \geq \Delta$

then find $\tau \geq 0$ so that $\|C(s_j + \tau p_j)\| = \Delta$,
and **return** $s_j + \tau p_j$

$r_{j+1} = r_j - \alpha_j B p_j$

if $\sqrt{r_{j+1}^T (C^T C)^{-1} r_{j+1}} / \sqrt{r_0^T (C^T C)^{-1} r_0} < \epsilon$ **then return** s_{j+1}

$\beta_{j+1} = r_{j+1}^T (C^T C)^{-1} r_{j+1} / r_j^T (C^T C)^{-1} r_j$

$p_{j+1} = (C^T C)^{-1} r_{j+1} + \beta_{j+1} p_j$

continue loop, after incrementing j

Algorithms 3.1 and 3.2 are equivalent and generate the same set of iterates. The first step s_1 of Algorithm 3.2 is again s_{cp} , and each residual r_j is the gradient of ϕ at s_j . If the trust region constraint is inactive and B is positive definite, then Algorithm 3.2 reduces to a *preconditioned* CG computation of the Newton step, with $C^T C$ acting as the preconditioning matrix (although this “preconditioner” does not necessarily make the problem better conditioned).

We are now ready to discuss the implementation of Algorithm 2.1.

4. Detailed description of the algorithm. We begin by applying the solution techniques of the previous section to the vertical and horizontal subproblems, paying close attention to the linear algebra costs. Following that, we discuss other components of the algorithm that also require a careful implementation.

4.1. Vertical step. Let us replace (2.4)–(2.5) by the equivalent problem

$$(4.1) \quad \min_{v \in \mathbf{R}^n} \quad c_k^T A_k^T v + \frac{1}{2} v^T A_k A_k^T v$$

$$(4.2) \quad \text{subject to} \quad \|v\| \leq \zeta \Delta_k,$$

and remember that we want the solution v_k to lie in the range space of A_k (to make it orthogonal to the horizontal step). To show that such a solution always exists, we note that the exact solution of (4.1)–(4.2) satisfies the equation

$$(4.3) \quad (A_k A_k^T + \rho I) v_* = -A_k c_k$$

for some $\rho \geq 0$ [20, p. 101]. If $\rho = 0$, then $v_* = -A_k (A_k^T A_k)^{-1} c_k$ is a solution, and it is clearly in the range space of A_k . On the other hand, if $\rho > 0$ then premultiplying both sides of (4.3) by Z_k^T gives $Z_k^T v_* = 0$, showing that v_* is in the range of A_k . Our implementation provides two options for computing an approximate vertical step v_k —the dogleg method and Steihaug’s method—and in both cases we ensure that v_k lies in the range space of A_k .

When we compare (4.1)–(4.2) with the general formulation (3.1), we see that $C = I$, $B = A_k A_k^T$ is positive semidefinite, $\Delta = \zeta \Delta_k$, and $g = A_k c_k$. Substituting into (3.7), we obtain the Cauchy point

$$v_{cp} = -\alpha A_k c_k,$$

where

$$\alpha = \min \left\{ \frac{\zeta \Delta_k}{\|A_k c_k\|}, \frac{\|A_k c_k\|^2}{(A_k c_k)^T A_k A_k^T (A_k c_k)} \right\}.$$

To formulate a dogleg method for (4.1)–(4.2) we need to define the Newton step. Since the matrix $A_k A_k^T$ has $n - m$ linearly independent null vectors, there is a whole manifold of minimizers for (4.1), each satisfying the linear equations (2.2). In our implementation we choose the shortest step, uniquely given by

$$(4.4) \quad v_n = -A_k (A_k^T A_k)^{-1} c_k.$$

Together, v_{cp} and v_n define a dogleg method for the vertical subproblem.

The main computational expense of the dogleg method lies in solving the linear system involving $A_k^T A_k$ to obtain v_n . We provide two strategies for performing

this calculation. One is to compute a sparse Cholesky factorization of $A_k^T A_k$ (using the subroutines of Ng and Peyton [30]), which gives an accurate answer (unless A_k is extremely ill conditioned), but could take $O(n^3)$ operations in the worst case. The alternative is to estimate v_n in (4.4) by the CG method (this is equivalent to Craig's method [10] applied to (4.1)), which can be a cheaper calculation, although less accurate. Preconditioners would clearly enhance the performance of this inner CG iteration, but none have been implemented yet in our software. Regardless of whether Cholesky or CGs are used for solving (4.4) we include the following safeguard: if $\|v_n\|$ is smaller than $\|v_{cp}\|$, then serious roundoff errors may have occurred in the computation of v_n and we revert to using just the Cauchy segment of the dogleg path. This is, in essence, our application of the dogleg method for the computation of the vertical step.

Steihaug's method as given in Algorithm 3.2 can also be applied to the vertical subproblem. The Hessian $B = A_k A_k^T$ is of size $n \times n$, but all search directions p_j in Algorithm 3.2 lie in the m -dimensional range space of A_k . Our test results indicate Steihaug's method sometimes performs better than the dogleg method and is certainly an alternative worth considering.

4.2. Horizontal step. For convenience let us write W_k for the Hessian of the Lagrangian or its limited memory BFGS approximation, and define $\bar{g}_k \equiv g_k + W_k v_k$ and $\bar{\Delta}_k^2 \equiv \Delta_k^2 - \|v_k\|^2$. Then the horizontal subproblem (2.9)–(2.10) can be written as

$$(4.5) \quad \min_{u \in \mathbf{R}^{n-m}} \quad \bar{g}_k^T Z_k u + \frac{1}{2} u^T Z_k^T W_k Z_k u$$

$$(4.6) \quad \text{subject to} \quad \|Z_k u\| \leq \bar{\Delta}_k.$$

This is a trust region subproblem with ellipsoidal constraint; in terms of the general formulation (3.1) we have $C = Z_k$, $B = Z_k^T W_k Z_k$, $g = Z_k^T \bar{g}_k$, $p = n$, and $q = n - m$. Using (3.7) we see that the Cauchy point is given by

$$(4.7) \quad u_{cp} = -\alpha \bar{u}_k, \quad \text{with} \quad \bar{u}_k \equiv (Z_k^T Z_k)^{-1} Z_k^T \bar{g}_k$$

and α determined by the following rule:

$$\begin{aligned} & \text{if } \bar{u}_k^T Z_k^T W_k Z_k \bar{u}_k \leq 0 \\ & \quad \text{then} \\ & \quad \alpha = \frac{\bar{\Delta}_k}{\|Z_k \bar{u}_k\|} \\ & \quad \text{else} \\ & \quad \alpha = \min \left\{ \frac{\bar{\Delta}_k}{\|Z_k \bar{u}_k\|}, \frac{\bar{g}_k^T Z_k \bar{u}_k}{\bar{u}_k^T Z_k^T W_k Z_k \bar{u}_k} \right\}. \end{aligned}$$

It will be seen in section 4.3 that directly computing $(Z_k^T Z_k)^{-1}$ can be very expensive, but the cost of multiplying Z_k and Z_k^T by a vector is much lower; therefore, we solve for $\bar{u}_k$ approximately by the CG method.

This means that u_{cp} is just an approximation to the Cauchy step, raising the question whether global convergence of the Byrd–Omojokun algorithm is still assured. Analysis similar to that in [6] shows that an approximation to u_{cp} is adequate for global convergence as long as the error in its computation tends to zero as fast as $Z_k^T \bar{g}_k$. Therefore, the stop tolerance in the CG iteration used for computing (4.7) is set to a small fraction of $\|Z_k^T \bar{g}_k\|$.

The horizontal step should make at least as much a reduction in (4.5) as the Cauchy step u_{cp} does, and preferably more in order to accelerate convergence. The dogleg method is only applicable to (4.5)–(4.6) in cases where W_k can be guaranteed positive definite, but since Steihaug's method can be applied for any W_k , we adopt it as our sole option for computing the horizontal step. To define Steihaug's method we only need to make the appropriate substitutions from (4.5)–(4.6) into Algorithm 3.2. This generates a set of iterates $\{u_j\}$ which estimate the solution of (4.5)–(4.6); the final iterate is then multiplied by Z_k to define the horizontal step.

A close examination reveals that this Steihaug iteration can be made more efficient by directly generating iterates and search directions of the form $Z_k u_j$ and $Z_k p_j$. To see this, note that the matrices B and C appearing in Algorithm 3.2 are given in this case by $B = Z_k^T W_k Z_k$ and $C = Z_k$. Thus, it is necessary to first multiply the quantities u_j (or s_j in the notation of Algorithm 3.2) and p_j by Z_k at every place they appear in the iteration. Instead of computing u_j and p_j themselves, we can work with $Z_k u_j$ and $Z_k p_j$ directly, saving the cost of multiplying by Z_k at every iteration of Steihaug's method. These ideas are implemented in Algorithm 4.1, where the symbols $(Zu)_j$ and $(Zp)_j$ denote the iterates and search directions in the desired form. By construction, all search directions $(Zp)_j$ lie in the null space of the constraint matrix, i.e., in the span of Z_k . Algorithm 4.1 also expresses the ellipsoidally scaled residual explicitly as a vector t_j , a quantity we shall examine more closely in a moment.

ALGORITHM 4.1. *Computation of the horizontal step*

Constants $\epsilon_{h_0} > 0$ and $\epsilon_h > 0$ are given
 Start with $(Zu)_0 = 0$ and $r_0 = -Z_k^T \bar{g}_k$
 Solve $(Z_k^T Z_k) t_0 = r_0$ for t_0 , and set $(Zp)_0 = Z_k t_0$
if $\sqrt{r_0^T t_0} < \epsilon_{h_0}$
 then use (4.7) and **return** the Cauchy step $Z_k u_{cp}$
loop, starting with $j = 0$
 if $(Zp)_j^T W_k (Zp)_j \leq 10^{-8} (Zp)_j^T (Zp)_j$
 then find τ so that $(Zu) = (Zu)_j + \tau (Zp)_j$ minimizes (4.5)
 and satisfies $\|(Zu)\| = \bar{\Delta}_k$, and **return** (Zu)
 $\alpha_j = r_j^T t_j / (Zp)_j^T W_k (Zp)_j$
 $(Zu)_{j+1} = (Zu)_j + \alpha_j (Zp)_j$
 if $\|(Zu)_{j+1}\| \geq \bar{\Delta}_k$
 then find $\tau \geq 0$ so that $\|(Zu)_j + \tau (Zp)_j\| = \bar{\Delta}_k$
 and **return** $(Zu)_j + \tau (Zp)_j$
 $r_{j+1} = r_j - \alpha_j Z_k^T W_k (Zp)_j$
 Solve $(Z_k^T Z_k) t_{j+1} = r_{j+1}$ for t_{j+1}
 if $\sqrt{r_{j+1}^T t_{j+1}} < \epsilon_h \sqrt{r_0^T t_0}$ **then return** $(Zu)_{j+1}$
 $\beta_{j+1} = r_{j+1}^T t_{j+1} / r_j^T t_j$
 $(Zp)_{j+1} = Z_k t_{j+1} + \beta_{j+1} (Zp)_j$
 continue loop, after incrementing j

As in the computation of the Cauchy step (4.7), we calculate t_{j+1} by approximately solving $(Z_k^T Z_k) t_{j+1} = r_{j+1}$ by the CG method. This inner loop is terminated when its residual becomes smaller than a fraction of $\epsilon_h \sqrt{r_0^T t_0}$, which is the stop tolerance in Algorithm 4.1. Since $r_0^T t_0$ is proportional to $\|Z_k^T g_k\|^2$, an analysis similar to that in [11] shows that this tolerance will give the algorithm a Q-quadratic asymptotic rate of convergence in the case when second derivatives are used for W_k .

Unscaled Steihaug iteration. The approximate solution to the horizontal subproblem found by Algorithm 4.1 turns out to be quite adequate for solving (1.1), but the calculation of the scaled residual $(Z_k^T Z_k)t_{j+1} = r_{j+1}$ is very time consuming in many problems. This CG subiteration is effectively a triply nested loop since Algorithm 4.1 lies within the main loop of Algorithm 2.1. Thus, we are motivated to search for less expensive alternatives for computing u_k that preserve the robustness of the trust region algorithm. Our most successful idea, which we call the *unscaled Steihaug iteration*, is to eliminate the matrix $(Z_k^T Z_k)^{-1}$ from Algorithm 4.1, thereby setting $t_{j+1} = r_{j+1}$. This can be viewed as preconditioning with the matrix $Z_k^T Z_k$; however, this “preconditioner” is not chosen for the usual purpose of improving the condition number of $Z_k^T W_k Z_k$ but only to reduce the cost of the iteration. Indeed, ignoring the matrix $(Z_k^T Z_k)^{-1}$ can be problematic, and must be done with certain precautions.

If $Z_k^T W_k Z_k$ is positive definite, and if the unscaled Steihaug iteration remains inside the scaled trust region $\|Zu\| \leq \bar{\Delta}$, then the solution obtained is simply the Newton step for (4.5). In this case the “preconditioner” has had no effect on the answer, and savings in computing time have been obtained. But if the unscaled Steihaug method terminates by reaching the trust region boundary, it is possible that the resulting step is poor compared with the solution obtained by the regular Steihaug iteration. The source of the difficulty is that, by removing the term $(Z_k^T Z_k)^{-1}$, the inequalities $\|(Zu)_{j+1}\| > \|(Zu)_j\|$ may no longer hold for all j . Therefore, an early iterate $(Zu)_j$ could leave the trust region and a later iterate $(Zu)_{j+s}$, giving a significantly lower value of (4.5), may subsequently return to the trust region. (It is still true that (4.5) is reduced monotonically by the sequence of steps $(Zu)_j$.)

To overcome this potential problem, we start with the unscaled Steihaug iteration but discard all the information generated if it leaves the scaled trust region, instead applying Algorithm 4.1. To minimize wasted effort, the unscaled Steihaug method is tried only if the last step d_k was accepted and Δ_{k+1} is the same as Δ_k . In particular, this ensures the method is tried as the iterates converge to a solution and the trust region becomes inactive. As a further precaution the stop test requires two successive residuals to be less than the tolerance ϵ_h . Numerical tests comparing both variations of the Steihaug iteration are given in section 5.1.

4.3. Computation of Z_k . Recall that Z_k is the $n \times (n - m)$ matrix spanning the manifold tangent to all constraints at x_k ; this means it satisfies $A_k^T Z_k = 0$. A numerically stable way of calculating Z_k from A_k is by means of the QR factorization, which ensures that the columns of Z_k are orthogonal. Although a sparse QR factorization appears feasible, we suspect that it may be costly in many cases and have deferred exploring this option to a later date. The idea we have pursued is one of direct elimination (see for example [20, p. 234]). First, partition A_k^T into

$$(4.8) \quad A_k^T = [B_k \ N_k],$$

where the $m \times m$ basis matrix B_k is nonsingular (for simplicity (4.8) assumes that the basis B_k is formed by the first m columns of A_k^T). Now define Z_k to be

$$(4.9) \quad Z_k = \begin{bmatrix} -B_k^{-1} N_k \\ I \end{bmatrix},$$

which satisfies $A_k^T Z_k = 0$, although the columns of Z_k are not orthogonal. Our software uses Harwell subroutine MA28 [15] to first choose m columns of A_k^T that

define the basis (MA28 can do this even though A_k^T is not square), and then compute the sparse LU factorization of B_k . MA28 selects the basis by considering both the resulting sparsity and the size of the pivots in the LU factorization of B_k . Forming Z_k explicitly could be very expensive since $B_k^{-1}N_k$ requires $n - m$ back-solves, and could destroy sparsity. Instead, the implicit representation (4.9) of Z_k is employed to compute matrix-vector products of the form $Z_k u$ and $Z_k^T x$ (Murtagh and Saunders use the same concept in [29]). For example, $Z_k u$ is obtained by computing $b = N_k u$, solving $B_k y = -b$ for y using the sparse LU factors of B_k , and appending u for the last components.

MA28 is designed to take advantage of situations where matrix values change but the sparsity pattern remains the same. This will be the case in our algorithm as long as we can keep the same m columns to serve as the basis matrix. At the beginning of the algorithm we do the costly work of choosing columns for B_0 and symbolically factoring it, and then use this same basis for subsequent iterates, saving many computations. But it is possible for iterates to change enough that B_k becomes very ill conditioned, in which case it is necessary to choose a different set of columns to define the basis. We have devised the following heuristic to determine when to change basis and refactor.

Heuristic I: Strategy for changing basis. When solving systems of the form $B_k y = b$ or $B_k^T y = b$ (which arise in the horizontal subproblem), we compute the ratio $\|b\|/\|y\|$. This is a number that lies somewhere between the largest and smallest singular values of B_k . During the lifetime of a particular choice of basis we remember the largest and smallest ratios seen, then divide these to estimate the condition number of the current B_k . If the estimate becomes greater than 10^2 , then on the next iteration we let MA28 try and choose a new basis and symbolically refactor.

To prevent this heuristic from interfering with good algorithm behavior we overrule it, and do not change basis, when the step is inside the trust region and is accepted by the algorithm. The performance of the heuristic is documented in section 5.1.

4.4. Limited memory approximations. Our software supplies an alternative when second derivatives are unavailable or prohibitively expensive to calculate. A limited memory BFGS (ℓ -BFGS) approximation to $\nabla_x^2 L_k$ is maintained and used in the role of W_k . The estimate is initialized with a scalar multiple of the identity matrix, then is updated implicitly as described in [27]. Only information reflecting the last t updates is kept, which requires $O(tn)$ storage locations and $O(tn)$ operations for updating W_k and multiplying W_k by a vector. Limited memory approximations give rise to a linear convergence rate, so there can be a significant loss in performance compared with using second derivatives. Previous research [22], [27] has shown that the choice $t \in [4, 10]$ gives good performance, and our testing suggests ℓ -BFGS is a viable option within the framework of this algorithm.

The software implementation uses the compact representations of limited memory BFGS matrices described in [4]. Every (implicit) BFGS update is of the form

$$(4.10) \quad W_{k+1} = W_k - \frac{W_k d_k d_k^T W_k}{d_k^T W_k d_k} + \frac{y_k y_k^T}{y_k^T d_k},$$

where $d_k = x_{k+1} - x_k$ and $y_k = (g_{k+1} - A_{k+1} \lambda_{k+1}) - (g_k - A_k \lambda_k)$. If $y_k^T d_k$ is not positive, we enforce this condition by modifying y_k according to Powell's damping strategy [36]:

if $d_k^T y_k < 0.2 d_k^T W_k d_k$
then $y_k \leftarrow \theta y_k + (1 - \theta) W_k d_k$,

where

$$\theta = \frac{0.8 d_k^T W_k d_k}{d_k^T W_k d_k - d_k^T y_k}.$$

There is also a danger that the last term in (4.10) becomes too large. To avoid this the update is not performed if

$$10^{-8} y_k^T y_k \geq y_k^T d_k.$$

4.5. Lagrange multipliers. At every new iterate x_{k+1} we compute least squares Lagrange multiplier estimates λ_{k+1} by solving

$$(4.11) \quad (A_{k+1}^T A_{k+1}) \lambda_{k+1} = A_{k+1}^T g_{k+1}.$$

This linear system is solved by either sparse Cholesky factorization or by the CG method. The accuracy of the Lagrange multipliers needs to be good enough to allow proper evaluation of the first-order KKT measure $\|g_k - A_k \lambda_k\|_\infty$. Therefore, when the CG method is used, the iteration is stopped when the residual in (4.11) becomes less than $10^{-2} * \max\{\|c_k\|_\infty, \|g_k - A_k \lambda_k\|_\infty\}$.

4.6. Merit function and choice of penalty parameter. The merit function is used to decide whether the step d_k makes sufficient progress toward the solution of problem (1.1). We follow the Byrd and Omojokun algorithm in [32] and use as merit function

$$(4.12) \quad \psi(x, \mu) = f(x) + \mu \|c(x)\|,$$

where $\mu > 0$ is called the penalty parameter; recall that $\|\cdot\|$ denotes the ℓ_2 norm. Since the last term in (4.12) is not squared, this merit function is not differentiable. It is also exact: in a neighborhood of a solution point x_* , and for $\mu > \|\lambda_*\|_\infty$, the minimizer of $\psi(x, \mu)$ is precisely x_* . We will see below that this merit function has the advantage that the ℓ_2 norm used to penalize the constraints is compatible with the ℓ_2 norm used in the formulation (2.4) of the vertical step.

We need to decide how the penalty parameter will be chosen at each iteration of the algorithm. As is commonly done (see, for example, [24], [37]) we first compute a step d_k and then choose μ large enough that d_k results in a reduction of the model. To be more precise, let us replace f by the model objective (2.1) and linearize the constraints in (4.12) to give the model merit function at x_k ,

$$\hat{\psi}_k(d, \mu) \equiv d^T g_k + \frac{1}{2} d^T W_k d + \mu \|A_k^T d + c_k\|,$$

where, as before, W_k denotes $\nabla_x^2 L_k$ or an ℓ -BFGS approximation to it. We now define the predicted reduction p-red in the model merit function $\hat{\psi}_k$ by

$$(4.13) \quad \begin{aligned} \text{p-red} &= \hat{\psi}_k(0, \mu) - \hat{\psi}_k(d_k, \mu) \\ &= -d_k^T g_k - \frac{1}{2} d_k^T W_k d_k + \mu (\|c_k\| - \|A_k^T v_k + c_k\|), \end{aligned}$$

where we have used (2.11) and the fact that $A_k^T Z_k = 0$. The term inside the parentheses is nonnegative because the dogleg or Steihaug methods used in the vertical step computation reduce the objective $\|A_k^T v + c_k\|$ from its initial value $\|c_k\|$ at $v = 0$. We now compute the trial value

$$(4.14) \quad \mu^+ = \max \left\{ \mu_k, 0.1 + \frac{d_k^T g_k + \frac{1}{2} d_k^T W_k d_k}{\|c_k\| - \|A_k^T v_k + c_k\|} \right\},$$

where μ_k is the penalty parameter at the previous iteration. It is clear from (4.13) that for this value of the penalty parameter the predicted reduction p_red is positive. (The term inside the parentheses in (4.13) can be zero when $v_k = 0$, which only happens when $c_k = 0$. In this case (4.14) is not used because p_red is made positive from the decrease that d_k makes in the objective of the horizontal subproblem (4.5).)

Before accepting the step, Algorithm 2.1 tests whether this choice of the penalty parameter leads to an actual decrease in the merit function (and not just in the model). We define the actual reduction a_red in the merit function by

$$(4.15) \quad \begin{aligned} \text{a\_red} &= \psi(x_k, \mu^+) - \psi(x_k + d_k, \mu^+) \\ &= f(x_k) - f(x_k + d_k) + \mu^+ (\|c(x_k)\| - \|c(x_k + d_k)\|). \end{aligned}$$

If a_red is sufficiently positive in the sense that $\text{a\_red} \geq \eta \text{p\_red}$, d_k is accepted and we set $\mu_{k+1} = \mu^+$; otherwise, the second-order correction procedure described in section 4.7 is invoked.

There is a danger of v_k being so small that the denominator of (4.14) approaches zero and μ^+ blows up, but this will normally happen only if the constraints are already nearly satisfied; in this case there is no point in further penalizing the constraints and we are justified in keeping $\mu^+ = \mu_k$. These guidelines for choosing the penalty parameter are sufficient for robust algorithm performance, but we have observed that the following modifications can often help the algorithm reach a solution faster.

Heuristic II: Update of penalty parameter. For certain problems with nonlinear constraints, convergence can be very slow because the penalty parameter is not increased enough. Formula (4.14) computes μ^+ using a linearization of the constraints, but if these are sufficiently nonlinear that $\|c(x_k)\| - \|c(x_k + d_k)\|$ is not well approximated by $\|c_k\| - \|A_k^T d_k + c_k\|$, then the step may increase the merit function and be rejected. In these cases it is advantageous to give more weight to constraint satisfaction by further increasing the penalty parameter. However, there is a risk that an unnecessarily large penalty parameter will cause iterates to follow the constraint manifold too closely, resulting in an excessive number of iterations for convergence. In the following, heuristic $k - s$, with $s \geq 1$, denotes the index of the most recent iterate at which the step d_{k-s} was accepted by the algorithm.

ALGORITHM 4.2. *Heuristic for modifying the penalty parameter*

Compute μ^+ using (4.14)

if $\mu^+ > \mu_k$ **and** $\mu^+ < 5\mu_k$ **and** $\mu_k > \mu_{k-s}$ **and** $\|c_k\| > \frac{1}{5}\|c_{k-s}\|$
and one or both of the last two steps were rejected

then $\mu^+ \leftarrow \min\{5\mu_k, \mu^+ + 25(\mu^+ - \mu_{k-s})\}$

if $\mu^+ = \mu_k$ **and** $\|v_k\| < \zeta \Delta_k / 10$ **and** $\|c_k\|_\infty < 10^4 \epsilon$

then $\mu^+ \leftarrow \max\{\mu_0, \bar{\mu}, \|\lambda_k\|\}$

(ϵ is the stop tolerance in Algorithm 2.1 and $\bar{\mu}$ is the second term inside the brackets in (4.14).)

The first **if** statement allows a further increase in the penalty parameter if not much progress ($\|c_k\| > \frac{1}{5}\|c_{k-s}\|$) has been made toward feasibility—a condition that

indicates that the constraint error is not being weighed strongly enough in the merit function—and if one of the last two steps failed, implying that the linearized constraint model is not a good fit. Safeguards are included to prevent the penalty parameter from changing too rapidly ($\mu^+ < 5\mu_k$) and to permit increases only when the penalty parameter is already on the rise ($\mu_k > \mu_{k-s}$). The second **if** statement permits reductions of the penalty parameter when iterates are close to satisfying the equality constraints; furthermore, the reduced μ^+ is never smaller than $\|\lambda_k\|$ (which is an estimate of the theoretical safe lower bound on μ_*) or smaller than the value $\bar{\mu}$ needed to assure that the step is a descent direction for the model merit function. In section 5.1 we present numerical results comparing this heuristic with the simpler rule (4.14).

4.7. Second-order correction. The merit function (4.12) is nondifferentiable and can give rise to the Maratos effect [28], which results in poor performance on some problems. The effect can be overcome by computing a second-order correction term (see [20, pp. 393–396] or [8] for details) and adding it to d_k , giving the new trial step

$$(4.16) \quad d_{\text{SOC}} = d_k - A_k(A_k^T A_k)^{-1} c(x_k + d_k).$$

The linear system involving $A_k^T A_k$ is solved just as in the computation of v_n in section 4.1: either by sparse Cholesky factorization or by CG. Computation of the second-order correction term is costly, so we only try d_{SOC} if the step d_k has been rejected, x_k is already nearly feasible, and $\|v_k\|$ is small compared with $\|Z_k u_k\|$. These conditions attempt to identify the occurrence of the Maratos effect and are much simpler than the rules given by Fletcher [19]. Combining this with the ideas of section 4.6, the final **if** statement of Algorithm 2.1 is more precisely defined by the following rules.

Compute μ^+ from Algorithm 4.2, a.red from (4.15), and p.red from (4.13)

```

if  $\frac{\text{a.red}}{\text{p.red}} \geq \eta$ 
  then  $x_{k+1} = x_k + d_k$ ,  $\mu_{k+1} = \mu^+$ ,  $\Delta_{k+1} \geq \Delta_k$ 
  else if  $\|v_k\| \leq 0.8\zeta\Delta_k$  and  $\|v_k\| \leq 0.1\|Z_k u_k\|$ 
    then compute  $d_{\text{SOC}}$  using (4.16)
      recalculate  $\text{a.red}$  with  $d_k \leftarrow d_{\text{SOC}}$ 
      if  $\frac{\text{a.red}}{\text{p.red}} \geq \eta$ 
        then  $x_{k+1} = x_k + d_{\text{SOC}}$ ,
           $\mu_{k+1} = \mu^+$ ,  $\Delta_{k+1} \geq \Delta_k$ 
        else  $x_{k+1} = x_k$ ,
           $\mu_{k+1} = \mu_k$ ,  $\Delta_{k+1} < \|d_k\|$ 
      else  $x_{k+1} = x_k$ ,  $\mu_{k+1} = \mu_k$ ,  $\Delta_{k+1} < \|d_k\|$ .

```

4.8. Modifying the trust region. We will now describe in detail how to update the trust region radius. Our numerical experience suggests that the following aggressive strategy often saves many iterations of the algorithm and is rarely harmful. When a step d_k (or d_{SOC}) is accepted we increase the trust radius according to the following rule:

```

if  $\frac{\text{a.red}}{\text{p.red}} \geq 0.9$ 
  then  $\Delta_{k+1} = \max\{10\|d_k\|, \Delta_k\}$ 
else if  $\frac{\text{a.red}}{\text{p.red}} \geq 0.3$ 
  then  $\Delta_{k+1} = \max\{2\|d_k\|, \Delta_k\}$ 
else  $\Delta_{k+1} = \Delta_k$  .

```

When the step is rejected, Δ_{k+1} is reduced to a fraction of the failed step length such that $\Delta_{k+1} \in [0.1 \|d_k\|, 0.5 \|d_k\|]$. The precise value is computed by assuming that the ratio of actual to predicted reduction is a linear function h of the step length $\|d\|$, satisfying $h(0) = 1$ and $h(\|d_k\|) = (\text{a\_red}/\text{p\_red})$, and then finding the value of $\|d\|$ where h equals η . Thus, the computation of Δ_{k+1} for a failed step is as follows:

```

Set  $\Delta_{k+1} = \frac{1 - \eta}{1 - (\text{a\_red}/\text{p\_red})} \|d_k\|$ 
if  $\Delta_{k+1} > 0.5 \|d_k\|$ 
    then  $\Delta_{k+1} \leftarrow 0.5 \|d_k\|$ 
else if  $\Delta_{k+1} < 0.1 \|d_k\|$ 
    then  $\Delta_{k+1} \leftarrow 0.1 \|d_k\|$  .

```

For other strategies of updating the trust region radius see [13, section 6.4.3] and [9].

4.9. Summary of the algorithm implementation. The previous discussion has provided a detailed description of our implementation of the Byrd–Omojokun algorithm, including several options for approximately solving the subproblems that arise. Below is an expanded description of Algorithm 2.1, which closely reflects the form of our code.

ALGORITHM ETR. *Trust region algorithm for equality constrained optimization*

Constants $\epsilon > 0$, and $\eta, \zeta \in (0, 1)$ are given

Choose x_0 and $\Delta_0, \mu_0 > 0$

loop, starting with $k = 0$

 Compute f_k, c_k, g_k , and A_k

if Heuristic I (see section 4.3) indicates the basis B_k for A_k is ill conditioned

then Find a new basis and its LU factors

else Compute the LU factors of the current basis

 (The null-space basis Z_k is now defined by (4.9))

 Compute multipliers λ_k by solving $(A_k^T A_k) \lambda_k = A_k^T g_k$

 Option 1: solve the linear system by Cholesky factorization

 Option 2: solve the system by CG

if $\|g_k - A_k \lambda_k\|_\infty < \epsilon$ **and** $\|c_k\|_\infty < \epsilon$ **then stop**

 Compute the vertical step v_k

 Option 1: use dogleg method, obtain the Newton step v_n by Cholesky

 Option 2: use dogleg method, obtain the Newton step v_n by CG

 Option 3: use Steihaug's method

 Compute W_k

 Option 1: set $W_k = \nabla_x^2 L_k(x_k, \lambda_k)$

 Option 2: update the limited memory (ℓ -BFGS) approximation W_k

 Compute the horizontal step u_k with the unscaled Steihaug iteration (section 4.2)

 Set $d_k = v_k + Z_k u_k$

 Compute μ^+ using Heuristic II as given in Algorithm 4.2

 Compute a_red from (4.15) and p_red from (4.13)

if $\frac{\text{a\_red}}{\text{p\_red}} \geq \eta$

then $x_{k+1} = x_k + d_k$, $\mu_{k+1} = \mu^+$, and expand trust region

else if $\|v_k\| \leq 0.8\zeta\Delta_k$ **and** $\|v_k\| \leq 0.1\|Z_k u_k\|$

then Compute the second-order correction d_{SOC} from (4.16):

 use the same option as was chosen to compute λ_k

Recalculate $\underline{a}_{\text{red}}$ (4.15) at d_{SOC}

if $\frac{\underline{a}_{\text{red}}}{\underline{p}_{\text{red}}} \geq \eta$

then $x_{k+1} = x_k + d_{\text{SOC}}$, $\mu_{k+1} = \mu^+$, expand trust region

else $x_{k+1} = x_k$, $\mu_{k+1} = \mu_k$, contract trust region

else $x_{k+1} = x_k$, $\mu_{k+1} = \mu_k$, and contract trust region

(The expansion and contraction of the trust region Δ_{k+1} is made according to the rules of section 4.8)

continue loop, after incrementing k

Let us reiterate our strategy for solving the linear systems involving the matrix $A_k^T A_k$ occurring in the algorithm. We provide two ways (Cholesky factorization and CG) of solving these linear systems, which occur in the dogleg component of the vertical subproblem v_k , in the computation of the Lagrange multipliers λ_k , and in the second-order correction step d_{SOC} . For these systems we are consistent in our choice of linear solver: if the vertical step is computed by Cholesky factorization, then the Cholesky factors are saved and used for solving the other systems; otherwise, CG is used to solve all three linear systems.

For testing purposes, the parameters occurring in the algorithm were selected as follows. The stop tolerance was $\epsilon = 10^{-5}$, and we set $\eta = 0.1$, $\zeta = 0.8$, and $\Delta_0 = 1$. The penalty parameter was always started at $\mu_0 = 1$. In the horizontal step computation (Algorithm 4.1) we used stop tolerances of $\epsilon_h = 10^{-2}$ and $\epsilon_{h_0} = 10^{-10}$. The computation of the horizontal Cauchy step in (4.7) requires approximately solving the system $Z_k^T Z_k$ by CG; this inner iteration was stopped when the residual was less than $\max\{10^{-10}, 10^{-8} * \min\{1, \|Z_k^T \bar{g}_k\|\}\}$. A more relaxed CG tolerance of $\max\{10^{-7}, 10^{-2} * \min\{1, r_0^T t_0\}\}$ was used to solve for t_{j+1} in Algorithm 4.1. When CG was used to obtain Lagrange multipliers λ_{k+1} , we stopped iterating when the residual became smaller than $10^{-2} * \max\{\|c_k\|_\infty, \|g_k - A_k \lambda_k\|_\infty\}$. When CG was used to obtain v_n , the residual had to be less than $\max\{10^{-8}, 10^{-2} * \min\{1, \|c_k\|\}\}$, while the more stringent tolerance $\max\{10^{-8}, 10^{-7} * \min\{1, \|c(x_k + d_k)\|\}\}$ was used to compute d_{SOC} .

5. Test results. We tested our software implementation on a set of difficult nonlinear equality constrained problems drawn from the CUTE collection [1]. The problems are divided into two groups, large and small, based on the number of unknowns. Table 5.1 presents a brief overview of the large problems. An asterisk by the problem name means the original CUTE problem has been modified for our use; for example, nonlinear inequalities may have been dropped or changed into equalities, or the problem parameters may have been altered to obtain a more ill conditioned or badly scaled problem. (All problems can be obtained from the authors in electronic form.) The number of nonzero elements in A is denoted by $\text{nnz}(A)$. The value of the objective at the solution is given under $f(x_*)$ to the number of significant figures that all optimization methods agreed upon. Note that in some problems $n = m$; in these we seek the root of a system of nonlinear equations. The CUTE software interface computes all derivatives by analytic formulae and provides a means of checking results by comparing them with solutions obtained with other software packages.

Our main concern in this paper is the development of an algorithm suitable for large-scale optimization; therefore, we begin by examining performance for the problems in Table 5.1 and defer the testing of small optimization problems until section 5.2. We first present numerical results that explore the performance of three options introduced in the previous sections: the use of the unscaled Steihaug iteration; Heuristic I—the strategy for automatically choosing a new basis; and Heuristic II—the aggres-

TABLE 5.1
Description of large test problem set.

Problem name	n	m	$\text{nnz}(A)$	Objective	Constraints	$f(x_*)$
GENHS28	300	298	894	nonlinear	linear	3.31481×10^1
HAGER2	10001	5001	15001	nonlinear	linear	4.3208×10^{-1}
HAGER3	8001	4001	12001	nonlinear	linear	1.4096×10^{-1}
ORTHREGA	2053	1024	7168	nonlinear	nonlinear	5.66143×10^2
ORTHREGC	1005	500	3500	nonlinear	nonlinear	1.87906×10^1
ORTHREGD	203	100	500	nonlinear	nonlinear	3.05079×10^1
*ORTHREGDS	203	100	500	nonlinear	nonlinear	4.037×10^1
*ORTHREGDM	4003	2000	10000	nonlinear	nonlinear	1.55533×10^2
*ORTHREGFB	1205	400	3200	nonlinear	nonlinear	1.62006×10^1
*OPTCTRL3	122	83	283	nonlinear	nonlinear	1.34806×10^3
*OPTCTRL6	122	83	283	nonlinear	nonlinear	2.04802×10^3
DTOC1ND	2998	2000	13976	nonlinear	nonlinear	2.37703×10^1
DTOC2	2998	2000	7988	nonlinear	nonlinear	4.9723×10^{-1}
DTOC3	14999	10000	34995	nonlinear	linear	2.3526×10^2
DTOC4	14999	10000	34995	nonlinear	nonlinear	2.868×10^0
DTOC5	9999	5000	14998	nonlinear	nonlinear	1.534×10^0
DTOC6	2001	1001	3001	nonlinear	nonlinear	1.7176×10^0
EIGENA2	110	55	1000	nonlinear	nonlinear	0.00000×10^0
EIGENC2	462	231	9261	nonlinear	nonlinear	0.00000×10^0
ARTIF	1002	1002	3002	-	nonlinear	-
BRATU3D	27000	27000	158712	-	nonlinear	-
BROYDNBD	5000	5000	34984	-	nonlinear	-
HTRODEQ	4656	4656	27361	-	nonlinear	-

* Test problem was created specifically for this paper.

sive strategy for updating the penalty parameter. After we have established that these options perform well on a carefully chosen sample of our test problems, we describe the performance of the fully developed algorithm on the whole test set. All our computations were performed in FORTRAN double precision on a Silicon Graphics workstation running IRIX 5.2 using a 150 MHz MIPS R4400 processor and 64Mbytes RAM.

5.1. Testing of heuristics. Let us begin by considering the unscaled Steihaug iteration introduced in section 4.2 to reduce the cost of the horizontal step computation. Since this iteration ignores the matrix $(Z_k^T Z_k)^{-1}$ it generally performs more steps than the regular Steihaug method. However, Table 5.2 shows that the many steps of the unscaled Steihaug iteration are so much cheaper to compute that there is still an overall savings in execution time. These experiments used the same stop tolerance $\epsilon_h = 10^{-2}$ for both iterations, although the preconditioned algorithm requires that two consecutive residuals be small enough before stopping, as discussed in section 4.2. Table 5.2 gives the results on a selection of problems that highlight the differences between the two approaches and considers the use of both exact Hessians and ℓ -BFGS approximations. The numbers in the table give the total CPU time for solving the problem (*CPU*) and the cumulative total number of horizontal step iterations (*h itr*) of Algorithm 4.1. (The numbers were obtained by using the fastest option for computing the vertical step.)

We observe that execution time nearly always decreases, sometimes dramatically, using the unscaled Steihaug iteration. Since the restrictions described in section 4.2 on when to use the unscaled Steihaug iteration ensure that its application does not alter the number of function and gradient evaluations needed to solve a problem, we performed all subsequent experiments with the unscaled Steihaug iteration.

TABLE 5.2
Comparison of unscaled Steihaug and regular Steihaug iterations.

Problem name	Exact Hessians				ℓ -BFGS approximations			
	Unscaled Steihaug		Steihaug		Unscaled Steihaug		Steihaug	
	<i>h itr</i>	<i>CPU</i>	<i>h itr</i>	<i>CPU</i>	<i>h itr</i>	<i>CPU</i>	<i>h itr</i>	<i>CPU</i>
HAGER2	11	18.7 sec	11	18.7 sec	13	19.3 sec	9	20.2 sec
HAGER3	12	14.4 sec	12	14.4 sec	16	14.3 sec	9	15.8 sec
ORTHREGA	128	168.9 sec	118	170.9 sec	485	275.0 sec	313	292.2 sec
ORTHREGC	179	23.3 sec	111	30.1 sec	424	42.8 sec	160	56.0 sec
ORTHRGDM	40	33.2 sec	16	35.7 sec	41	42.6 sec	19	46.9 sec
ORTHRGFB	71	27.7 sec	71	27.7 sec	213	32.6 sec	128	37.3 sec
DTOC1ND	33	39.0 sec	26	41.6 sec	83	85.8 sec	60	93.5 sec
DTOC2	7139	398.3 sec	331	949.9 sec	18008	1311.2 sec	962	3023.3 sec
DTOC5	11	78.6 sec	10	79.3 sec	14	105.7 sec	9	104.7 sec
DTOC6	243	134.3 sec	246	139.1 sec	321	206.8 sec	314	201.8 sec
EIGENC2	582	38.6 sec	408	101.3 sec	5022	234.0 sec	2039	506.7 sec

TABLE 5.3
Performance of strategies for choosing a basis.

Problem name	Never change basis				Always new basis				Heuristic I			
	<i>nb</i>	<i>bas</i>	<i>hor</i>	<i>CPU</i>	<i>nb</i>	<i>bas</i>	<i>hor</i>	<i>CPU</i>	<i>nb</i>	<i>bas</i>	<i>hor</i>	<i>CPU</i>
ORTHREGA	1	16%	67%	164.3	20	67%	17%	194.6	15	63%	20%	168.9
ORTHREGC	1	3%	93%	165.0	15	40%	35%	27.9	4	24%	48%	23.3
ORTHRGDM	1	70%	14%	33.1	7	84%	7%	57.9	1	70%	14%	33.2
ORTHRGFB	1	2%	95%	242.5	15	38%	36%	27.2	3	17%	57%	27.7
DTOC6	1	8%	89%	141.7	25	9%	87%	135.5	21	9%	88%	134.3
EIGENC2	1	16%	76%	38.8	10	32%	63%	48.7	1	16%	76%	38.6

Let us now test Heuristic I—the strategy for choosing a new basis with Harwell subroutine MA28 described in section 4.3. We compare it with the alternative strategies of always using the same basis and always letting MA28 choose a basis freely. We expect the last strategy to produce a better conditioned basis and therefore a better-conditioned horizontal subproblem (4.5)–(4.6). Heuristic I seeks to balance the cost of using MA28 at every new iterate against the cost of solving badly conditioned horizontal subproblems.

Table 5.3 shows a comparison between the three strategies for those test problems in which it made a difference. (In the other problems either the basis chosen by MA28 never varied for the three strategies, or the relative cost of executing MA28 was insignificant.) For each strategy we list the number of times that a new basis was computed (*nb*), the percent of execution time spent by MA28 choosing and factoring a basis (*bas*), the percent of execution time spent computing the horizontal step, and the total number of CPU seconds used to solve the problem (*CPU*). All problems were run using exact Hessian information and the fastest option for computing v_k .

The data clearly shows a substantial tradeoff between the time spent finding a good basis and the time needed to solve the horizontal subproblem. Table 5.3 suggests that Heuristic I adequately controls the condition number of the basis and results in an acceptable computing time. Therefore, even though our sample of results is small, we have incorporated it into our code.

Next we study the performance of Heuristic II: the strategy for updating the penalty parameter described in section 4.6. Table 5.4 compares the number of function and constraint evaluations required for convergence using first the updating rule

TABLE 5.4
Comparison of two strategies for updating the penalty parameter.

Problem name	Exact Hessians		ℓ -BFGS approximations	
	Rule (4.14)	Heuristic II	Rule (4.14)	Heuristic II
ORTHREGA	70	44	138	115
ORTHREGC	36	26	93	73
ORTHREGFB	28	27	60	53
OPTCTRL3	7	7	64	63
OPTCTRL6	9	8	150	164
DTOC1ND	7	7	17	17
DTOC3	6	6	10	10
DTOC4	5	5	7	7
DTOC5	6	6	8	8
DTOC6	> 1000	43	> 1000	164

(4.14) and then Heuristic II, as given in Algorithm 4.2. We present results only for those problems for which the two strategies differed, that is, problems in which μ^+ is increased by Algorithm 4.2 at some iteration. (Again, each problem was run using the fastest option for computing v_k .)

The results in Table 5.4 show that Heuristic II is significantly beneficial for several problems while degrading performance in only one instance (OPTCTRL6). Our sample of results is again quite small, but since the heuristic appears to work rather well, it has been incorporated into the code.

5.2. Comprehensive tests. We are now ready to describe the performance of the algorithm on the full set of test problems. The values of all tolerances and parameters used by the algorithm during testing are given at the end of section 4.9. First we give results for many small nonlinear problems to test the robustness of the algorithm. Table 5.5 lists the number of unknowns (n) and constraints (m) in each problem, the number of times that the function and constraints were evaluated to obtain a solution (f, c), and the number of evaluations of gradients (g, A). As in Table 5.1, an asterisk by the problem name means the original CUTE problem has been modified for our use. We report the best results for the algorithm from among the three different options for computing the vertical step. Note that in problems where $n = m$ there is no horizontal subproblem and W_k plays no role in the algorithm.

The results given in Table 5.5 suggest that our implementation of the Byrd–Omojokun algorithm is robust. All the problems were solved successfully using the default algorithm parameters described in section 4.9, except for HS104LNP and HS111LNP. These two problems originally contained bound constraints that were inactive at the solution. We deleted the bounds, but to prevent iterates from straying into a region where f was not defined we had to limit the maximum growth of the trust region to a factor of 2 instead of 10 (see section 4.8).

To further test robustness of the algorithm we duplicated the experiments described in [43], in which problems HS6, HS7, HS8, HS9, HS26, HS27, HS39, HS40, HS42, HS60b, HS77, HS78, and HS79 were posed with a set of more difficult starting points. Out of 102 different test cases our algorithm (using exact Hessians and trying all three options for computing the vertical step) failed to reach a solution in seven instances. The failures were caused by rank deficiency in the constraint Jacobian A_k . Thus our code appears to be extremely robust, although it could be improved by better handling linear dependencies among the constraints.

TABLE 5.5
Performance on small test problems.

Problem	n	m	Exact Hessians		ℓ -BFGS approximations	
			Number f,c	Number g,A	Number f,c	Number g,A
BT1	2	1	51	18	12	8
BT2	3	1	13	13	16	16
BT4	3	2	7	7	12	11
BT5	3	2	7	7	10	10
BT6	5	2	17	13	19	15
BT7	5	3	36	22	361	170
BT8	5	2	10	10	13	13
BT10	2	2	7	7	no Hessian when $n = m$	
BT11	5	3	7	7	12	12
BT12	5	3	6	6	9	7
HS6	2	1	30	13	20	12
HS7	2	1	8	7	15	11
HS8	2	2	6	6	no Hessian when $n = m$	
HS9	2	1	6	4	6	6
HS26	3	1	16	16	42	24
HS27	3	1	39	18	25	17
HS39	4	2	17	13	26	19
HS40	4	3	4	4	7	5
HS42	4	2	5	5	8	8
*HS60b	3	1	6	6	13	11
HS77	5	2	11	9	23	16
HS78	5	3	5	5	8	7
HS79	5	3	6	6	12	12
*HS100LNP	7	2	11	8	18	15
*HS104LNP	8	4	12	11	23	21
*HS111LNP	10	3	11	11	30	24
METHANB8	31	31	4	4	no Hessian when $n = m$	
METHANL8	31	31	6	6	no Hessian when $n = m$	

* Test problem was created specifically for this paper.

Let us move on to our main goal in this section, testing algorithm performance on the large-scale problems of Table 5.1. Table 5.6 presents results using exact second derivatives to define the Hessian of the Lagrangian W_k and Table 5.7 the results using ℓ -BFGS approximations. Both tables characterize performance by listing the number of times that the function and constraints were evaluated (f, c), the number of evaluations of the gradients (g, A), and the total CPU time. Data is presented for all three methods for computing the vertical step: dogleg using the sparse Cholesky factorization of $A_k^T A_k$, Steihaug's method, and dogleg using CG to obtain the point v_n in (4.4). Since W_k plays no role in problems where $n = m$, those problems are omitted from Table 5.7.

In problems HAGER2, HAGER3, DTOC3, DTOC4, and DTOC5 the constraint Jacobian A_k is ill conditioned and large; consequently, the CG iterations used in the Steihaug and dogleg-CG options for computing v_k were very inefficient. On the other hand, a great amount of fill-in occurs during the formation or Cholesky factorization of $A_k^T A_k$ in the ORTHREG family of problems and BRATU3D. In three instances the fill-in even exceeded the 100Mbyte virtual memory workspace of our workstation. As these problems illustrate, the choice of iterative versus direct methods for computing the vertical step depends on the nature of the problem's constraints and can be critical to performance. We therefore make both options available to the user in our software.

The results of Table 5.7 indicate that using limited memory approximations for second derivatives gives rise to a slower algorithm that needs more function and first

TABLE 5.6
Performance results using exact Hessians.

Problem	v_k by dogleg-Cholesky			v_k by Steihaug			v_k by dogleg-CG		
	f,c	g,A	CPU time	f,c	g,A	CPU time	f,c	g,A	CPU time
GENHS28	4	4	0.2 sec	6	6	0.4 sec	6	6	0.4 sec
HAGER2	5	5	18.7 sec	5	5	884.7 sec	6	6	649.8 sec
HAGER3	5	5	14.4 sec	5	5	479.4 sec	6	6	399.5 sec
ORTHREGA	$A^T A$ too dense			42	19	164.2 sec	44	20	168.9 sec
ORTHREGC	53	20	119.6 sec	26	15	23.3 sec	53	20	41.1 sec
ORTHREGD	14	10	1.4 sec	8	8	0.6 sec	15	10	1.6 sec
ORTHRGDM	$A^T A$ too dense			8	8	37.9 sec	7	7	33.2 sec
ORTHRGFB	32	17	63.7 sec	27	15	27.7 sec	32	17	36.6 sec
OPTCTRL3	7	7	0.6 sec	7	7	0.8 sec	7	7	0.8 sec
OPTCTRL6	8	8	0.7 sec	9	9	1.1 sec	9	9	1.1 sec
DTOC1ND	7	7	39.0 sec	7	7	42.5 sec	7	7	42.2 sec
DTOC2	6	6	398.3 sec	7	7	502.1 sec	6	6	486.2 sec
DTOC3	6	6	68.9 sec	7	7	2505.3 sec	7	7	2991.1 sec
DTOC4	5	5	292.3 sec	12	9	6368.6 sec	5	5	3752.8 sec
DTOC5	6	6	78.6 sec	21	14	3190.6 sec	6	6	1081.1 sec
DTOC6	43	25	134.3 sec	27	18	171.9 sec	43	25	230.8 sec
EIGENA2	4	4	0.2 sec	4	4	0.2 sec	4	4	0.2 sec
EIGENC2	14	10	41.4 sec	14	10	38.7 sec	14	10	38.6 sec
ARTIF	10	10	1.0 sec	10	10	6.2 sec	14	11	14.7 sec
BRATU3D	factors of $A^T A$ too dense			5	5	216.5 sec	5	5	281.6 sec
BROYDNBD	$A^T A$ rank deficient			8	8	6.7 sec	did not converge		
HTRODEQ	12	10	36.3 sec	did not converge			did not converge		

TABLE 5.7
Performance results using ℓ -BFGS approximations.

Problem	v_k by dogleg-Cholesky			v_k by Steihaug			v_k by dogleg-CG		
	f,c	g,A	CPU time	f,c	g,A	CPU time	f,c	g,A	CPU time
GENHS28	8	8	0.3 sec	9	9	0.6 sec	9	9	0.6 sec
HAGER2	8	8	18.9 sec	8	8	1452.1 sec	8	8	937.4 sec
HAGER3	8	8	14.2 sec	8	8	803.4 sec	8	8	582.4 sec
ORTHREGA	$A^T A$ too dense			95	41	189.8 sec	115	49	215.3 sec
ORTHREGC	118	47	256.8 sec	73	36	43.0 sec	100	41	57.4 sec
ORTHREGD	12	11	1.1 sec	11	11	0.6 sec	13	12	0.7 sec
ORTHRGDM	$A^T A$ too dense			9	9	43.0 sec	9	9	41.7 sec
ORTHRGFB	28	21	64.6 sec	53	25	26.6 sec	46	24	29.5 sec
OPTCTRL3	63	45	2.1 sec	65	58	4.4 sec	68	47	3.7 sec
OPTCTRL6	did not converge			191	187	12.0 sec	164	154	9.7 sec
DTOC1ND	17	16	84.1 sec	22	18	105.7 sec	24	18	105.9 sec
DTOC2	102	93	1034.0 sec	195	99	3140.5 sec	101	88	2526.3 sec
DTOC3	10	8	70.3 sec	9	9	4402.1 sec	10	8	4433.4 sec
DTOC4	7	7	404.3 sec	7	7	4699.6 sec	7	7	6144.5 sec
DTOC5	8	8	103.2 sec	8	8	1795.1 sec	8	8	1610.7 sec
DTOC6	164	88	197.8 sec	89	52	364.2 sec	200	96	567.7 sec
EIGENA2	4	4	0.2 sec	4	4	0.2 sec	4	4	0.2 sec
EIGENC2	208	192	230.4 sec	217	203	193.6 sec	235	212	218.4 sec

derivative evaluations. This demonstrates the advantage of exploiting exact Hessian information. However, second derivatives are difficult to obtain in some problems, and then the availability of compact ℓ -BFGS matrices is a valuable algorithm feature.

We monitored the speed of convergence of the algorithm, and when second derivatives were used, all the problems exhibited quadratic contraction in the error during the final iterations. The ORTHREG family of problems have the kind of constraint curvature that gives rise to the Maratos effect, and our second-order corrections were

successful in maintaining a quadratic convergence rate. When ℓ -BFGS approximations were used for W_k the rate of convergence was linear, as expected.

The algorithm failed to converge on problems BROYDNBD and HTRODEQ for some of the vertical subproblem options. This is because the constraint Jacobian A in these problems is nearly rank deficient, preventing us from reducing constraint infeasibility beyond a certain point. We constructed problem ORTHRGDS (see Table 5.1) to investigate the limits of algorithm robustness when A_k becomes rank deficient. By a judicious choice of the starting point we can make the algorithm converge to a solution point where any desired number of the columns of A_* vanish. As we increased the number of zero columns, the algorithm took more and more CPU time to find a solution satisfying the stop tolerance of $\epsilon = 10^{-5}$. When more than 5 columns of A_* vanished at the solution, the algorithm was unable to meet this stop tolerance. There is clearly a great need for modifying the algorithm so that it can cope with linearly dependent constraint gradients, but this is a task that we have not yet addressed.

We modified the original OPTCNTRL problem so as to introduce ill conditioning by adding a quadratic term to the objective function f . The resulting problems, OPTCTRL3 and OPTCTRL6, have Hessian matrices with condition numbers of order 10^3 and 10^6 , respectively. In these problems the merit function penalty parameter becomes very large, but the algorithm is nevertheless able to reach the solution, except for one case: when ℓ -BFGS approximations are used with the dogleg-Cholesky option, rounding errors prevent the stop tolerance $\epsilon = 10^{-5}$ from being satisfied.

A detailed analysis of other test results, and a discussion of the properties of the test problems is given by Plantenga [33].

We also solved the large test problems with LANCELOT A (7/94) [9], using second derivatives and the same stop tolerance of 10^{-5} (specified for both `constraint-accuracy-required` and `gradient-accuracy-required`). The `infinity-norm-trust-region` was applied, and an `exact-cauchy-point-required`. LANCELOT provides a variety of preconditioners to choose from, and the results we state in Table 5.8 are the best performance by LANCELOT out of a set of 11 preconditioners tried. More specifically, for each problem we report the fewest function evaluations obtained over the set of all preconditioning options and the smallest execution time and note that the best performance with respect to these two measures was often obtained by different options. The results for our code were tabulated in the same manner. We do this because in some applications a user may want to minimize function evaluations, while in others execution time may be the top priority.

We certainly do not yet consider our software to be as general or robust as LANCELOT, and our tests show that it sometimes requires much more CPU time. But the data in Table 5.8 indicate that our algorithm has great promise; perhaps the most encouraging statistic is the smaller number of function and gradient evaluations required by our algorithm in these tests. In the next section we mention various possibilities for decreasing the CPU cost of an iteration.

6. Final remarks. During the last ten years much work has been devoted to the development of trust region methods for equality constrained optimization based on sequential quadratic programming ([42], [7], [38], [5], [12], [16], [17], [25], [43], [44], [46]). However, to our knowledge, none of these methods has received a careful implementation, and it was unknown to us if they would perform well in practice. We therefore undertook the task of implementing one of these methods and observing its performance both in the large-scale case and in the solution of ill-conditioned problems. We have chosen the trust region method of Byrd [2] and Omojokun [32] because

TABLE 5.8
Performance comparison with LANCELOT A (7/94) .

Problem	LANCELOT			Algorithm ETR		
	f,c	g,A	CPU time	f,c	g,A	CPU time
GENHS28	6	7	0.6 sec	4	4	0.2 sec
HAGER2	8	9	21.1 sec	5	5	18.7 sec
HAGER3	8	9	19.2 sec	5	5	14.4 sec
ORTHREGA	118	107	97.4 sec	42	19	164.2 sec
ORTHREGC	30	28	11.8 sec	26	15	23.3 sec
ORTHREGD	44	40	1.4 sec	8	8	0.6 sec
ORTHRGDM	29	26	26.0 sec	7	7	33.2 sec
ORTHRGFB	42	38	10.9 sec	27	15	27.7 sec
OPTCTRL3	37	38	0.8 sec	7	7	0.6 sec
OPTCTRL6	46	47	1.1 sec	8	8	0.7 sec
DTOC1ND	15	15	21.2 sec	7	7	39.0 sec
DTOC2	20	21	17.9 sec	6	6	398.3 sec
DTOC3	22	23	76.4 sec	6	6	68.9 sec
DTOC4	13	14	54.2 sec	5	5	292.3 sec
DTOC5	21	22	41.0 sec	6	6	78.6 sec
DTOC6	54	55	19.3 sec	27	18	134.3 sec
EIGENA2	13	14	0.5 sec	4	4	0.2 sec
EIGENC2	174	148	62.0 sec	14	10	38.6 sec
ARTIF	23	17	3.3 sec	10	10	1.0 sec
BRATU3D	3	4	169.2 sec	5	5	216.5 sec
BROYDNBD	12	12	16.0 sec	8	8	6.7 sec
HTRODEQ	did not converge			12	10	36.3 sec

we feel that it is well suited for large problems and because of its attractive convergence properties. Our implementation can incorporate second derivative information without an explicit Hessian matrix—only Hessian-vector products are needed. Alternatively, the software can supply an ℓ -BFGS Hessian approximation. Our numerical experience with a small but challenging set of test problems has convinced us that this algorithm is quite effective and merits further development.

The following refinements and options that we plan to investigate may significantly improve its efficiency:

1. Applying preconditioners to the many subproblems solved by CG iterations.
2. Replacing the system $Z_k^T Z_k$ in the horizontal subproblem with a full-space orthogonal projection, based on the identity $Z(Z^T Z)^{-1} Z^T = I - A(A^T A)^{-1} A^T$. Preliminary tests indicate this significantly reduces computing time when the Cholesky factorization of $A^T A$ can be obtained cheaply and accurately [26].
3. Using a sparse QR factorization of A_k to compute Z_k . This eliminates the difficulties associated with ill-conditioned bases in (4.8). The QR factorization may not always be economical, but it is likely to be very effective on some problems.
4. Solving linear systems involving $A_k^T A_k$ by the augmented system approach [14], [21]. This provides an interesting alternative to the normal equations and extends the class of problems for which a direct solution of the vertical problem is practical.
5. Computing the horizontal step more efficiently when using ℓ -BFGS approximations. Our current code approximates the full Hessian of the Lagrangian W_k , projects it onto the tangent space of the constraints, and applies Steihaug's method via Algorithm 4.1. The solution of this subproblem takes up to 60% of total execution time, far too high given that the ℓ -BFGS approximation is not very accurate. A promising alternative currently under investigation [31] is to solve the horizontal subproblem using a dual method.

6. Our current implementation requires that the matrices A_k of constraint gradients be of full rank, but this is not intrinsic to the Byrd–Omojokun approach. An important enhancement of our code will be the development of solvers that do not require this full-rank assumption.

But the most pressing need is to adapt the algorithm so that it can accept inequality constraints. There are various ways of doing this, and they are currently the subject of investigation [34], [3].

Acknowledgments. The authors would like to express their gratitude to Richard Byrd for his insights. We also thank the referees and the associate editor for many valuable comments and constructive suggestions.

REFERENCES

- [1] I. BONGARTZ, A. R. CONN, N. I. M. GOULD, AND PH. L. TOINT, *CUTE: Constrained and unconstrained testing environment*, ACM Trans. Math. Software, 21 (1995), pp. 123–160.
- [2] R. H. BYRD, *Robust trust region methods for constrained optimization*, Third SIAM Conference on Optimization, Houston, TX, May 1987.
- [3] R. H. BYRD, J. CH. GILBERT, AND J. NOCEDAL, *A Trust Region Method Based on Interior Point Techniques for Nonlinear Programming*, Tech. report OTC 96/02, Optimization Technology Center, Northwestern University, Evanston, IL, 1996.
- [4] R. H. BYRD, J. NOCEDAL, AND R. B. SCHNABEL, *Representations of quasi-Newton matrices and their use in limited memory methods*, Math. Programming, Ser. A, 63 (1994), pp. 129–156.
- [5] R. H. BYRD, R. B. SCHNABEL, AND G. A. SCHULTZ, *A trust region algorithm for nonlinearly constrained optimization*, SIAM J. Numer. Anal., 24 (1987), pp. 1152–1170.
- [6] R. G. CARTER, *On the global convergence of trust region algorithms using inexact gradient information*, SIAM J. Numer. Anal., 28 (1991), pp. 251–265.
- [7] M. R. CELIS, J. E. DENNIS, AND R. A. TAPIA, *A trust region strategy for nonlinear equality constrained optimization*, in Numerical Optimization 1984, P. T. Boggs, R. H. Byrd, and R. B. Schnabel, eds., SIAM, Philadelphia, 1985, pp. 71–82.
- [8] T. F. COLEMAN AND A. R. CONN, *Nonlinear programming via an exact penalty function: Global analysis*, Math. Programming, 24 (1982), pp. 137–161.
- [9] A. R. CONN, N. I. M. GOULD, AND PH. L. TOINT, *LANCELOT: A Fortran Package for Large-Scale Nonlinear Optimization (Release A)*, Springer Ser. Comput. Math., Springer-Verlag, Berlin, 1992.
- [10] E. J. CRAIG, *The N-step iteration procedure*, J. Math. Phys., 34 (1955), pp. 65–73.
- [11] R. S. DEMBO, S. C. EISENSTAT, AND T. STEIHAUG, *Inexact Newton methods*, SIAM J. Numer. Anal., 19 (1982), pp. 400–408.
- [12] J. E. DENNIS, M. EL-ALEM, AND M. C. MACIEL, *A Global Convergence Theory for General Trust-Region-Based Algorithms for Equality Constrained Optimization*, Tech. report 92-28, Dept. of Mathematical Sciences, Rice University, Houston, TX, 1992.
- [13] J. E. DENNIS JR. AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1983.
- [14] I. S. DUFF AND J. K. REID, *A comparison of some methods for the solution of sparse overdetermined systems of linear equations*, J. Inst. Maths. Applics., 17 (1976), pp. 267–280.
- [15] I. S. DUFF AND J. K. REID, *Some design features of a sparse matrix code*, ACM Trans. Math. Software, 5 (1979), pp. 18–35.
- [16] M. M. EL-ALEM, *A global convergence theory for the Celis-Dennis-Tapia trust region algorithm for constrained optimization*, SIAM J. Numer. Anal., 28 (1991), pp. 266–290.
- [17] M. M. EL-ALEM, *A Robust Trust-Region Algorithm with a Non-monotonic Penalty Parameter Scheme for Constrained Optimization*, Tech. report, Dept. of Mathematical Sciences, Rice University, Houston, TX, 1993.
- [18] R. FLETCHER, *An ℓ_1 penalty method for nonlinear constraints*, in Numerical Optimization 1984, P. T. Boggs, R. H. Byrd, and R. B. Schnabel, eds., SIAM, Philadelphia, 1985, pp. 26–40.
- [19] R. FLETCHER, *Second order corrections for nondifferentiable optimization*, in Numerical Analysis, Dundee 1981, G. A. Watson, ed., Lecture Notes in Math. 912, Springer-Verlag, Berlin, 1982.
- [20] R. FLETCHER, *Practical Methods of Optimization*, 2nd ed., Wiley & Sons, Chichester, UK, 1990.

- [21] R. FOURER AND S. MEHROTRA, *Solving symmetric indefinite systems in an interior-point method for linear programming*, Math. Programming, Ser. B, 62 (1993), pp. 15–39.
- [22] J. CH. GILBERT AND C. LEMARÉCHAL, *Some numerical experiments with variable-storage quasi-Newton algorithms*, Math. Programming, Ser. B, 45 (1989), pp. 407–435.
- [23] P. E. GILL, W. MURRAY, AND M. A. SAUNDERS, *SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization*, Numerical Analysis Report 96-2, Dept. of Mathematics, University of California, San Diego, La Jolla, CA, 1996.
- [24] S. P. HAN, *A globally convergent method for nonlinear programming*, J. Optim. Theory Appl., 22 (1977), pp. 297–309.
- [25] M. HEINKENSCHLOSS, *On the solution of a two-ball trust region subproblem*, Math. Programming, Ser. A, 64 (1994), pp. 249–276.
- [26] M. E. HRIBAR, *Large-Scale Constrained Optimization*, Ph.D. thesis, Dept. of Electrical Engineering and Computer Science, Northwestern University, Chicago, IL, 1996.
- [27] D. C. LIU AND J. NOCEDAL, *On the limited memory BFGS method for large scale optimization*, Math. Programming, Ser. B, 45 (1989), pp. 503–528.
- [28] N. MARATOS, *Exact Penalty Function Algorithms for Finite Dimensional and Control Optimization Problems*, Ph.D. thesis, University of London, UK, 1978.
- [29] B. A. MURTAGH AND M. A. SAUNDERS, *Large-scale linearly constrained optimization*, Math. Programming, 14 (1978), pp. 41–72.
- [30] E. G. NG AND B. W. PEYTON, *Block sparse Cholesky algorithms on advanced uniprocessor computers*, SIAM J. Sci. Statist. Comp., 14 (1993), pp. 1034–1056.
- [31] J. NOCEDAL AND T. D. PLANTENGA, *Solving KKT systems using limited memory BFGS matrices*, manuscript.
- [32] E. O. OMOJOKUN, *Trust Region Algorithms for Optimization with Nonlinear Equality and Inequality Constraints*, Ph.D. thesis, Dept. of Computer Science, University of Colorado, Boulder, 1989.
- [33] T. D. PLANTENGA, *Large-Scale Nonlinear Constrained Optimization Using Trust Regions*, Ph.D. thesis, Dept. of Electrical Engineering and Computer Science, Northwestern University, Chicago, IL, 1994.
- [34] T. D. PLANTENGA, *A trust region method for nonlinear programming based on primal interior-point techniques*, SIAM J. Sci. Comput., to appear.
- [35] M. J. D. POWELL, *A hybrid method for nonlinear equations*, in Numerical Methods for Nonlinear Algebraic Equations, P. Rabinowitz, ed., Gordon and Breach, London, UK, 1970, pp. 87–114.
- [36] M. J. D. POWELL, *The convergence of variable metric methods for nonlinearly constrained optimization calculations*, in Nonlinear Programming 3, O. Mangasarian, R. Meyer, and S. Robinson, eds., Academic Press, New York, London, 1978, pp. 27–63.
- [37] M. J. D. POWELL AND Y. YUAN, *A recursive quadratic programming algorithm that uses differentiable exact penalty functions*, Math. Programming, 35 (1986), pp. 265–278.
- [38] M. J. D. POWELL AND Y. YUAN, *A trust region algorithm for equality constrained optimization*, Math. Programming, Ser. A, 49 (1991), pp. 189–211.
- [39] G. A. SCHULTZ, R. B. SCHNABEL, AND R. H. BYRD, *A family of trust-region-based algorithms for unconstrained minimization with strong global convergence properties*, SIAM J. Numer. Anal., 22 (1985), pp. 47–67.
- [40] T. STEihaug, *The conjugate gradient method and trust regions in large scale optimization*, SIAM J. Numer. Anal., 20 (1983), pp. 626–637.
- [41] PH. L. TOINT, *Towards an efficient sparsity exploiting Newton method for minimization*, in Sparse Matrices and Their Uses, I. S. Duff, ed., Academic Press, New York, 1981, pp. 57–87.
- [42] A. VARDI, *A trust region algorithm for equality constrained minimization: Convergence properties and implementation*, SIAM J. Numer. Anal., 22 (1985), pp. 575–591.
- [43] K. A. WILLIAMSON, *A Robust Trust Region Algorithm for Nonlinear Programming*, Ph.D. thesis, Dept. of Mathematical Sciences, Rice University, Houston, TX, 1991.
- [44] Y. YUAN, *On a subproblem of trust region algorithms for constrained optimization*, Math. Programming, Ser. A, 47 (1990), pp. 53–63.
- [45] J. Z. ZHANG AND D. T. ZHU, *Projected quasi-Newton algorithm with trust region for constrained optimization*, J. Optim. Theory Appl., 67 (1990), pp. 369–393.
- [46] Y. ZHANG, *Computing a Celis-Dennis-Tapia trust region step for equality constrained optimization*, Math. Programming, Ser. A, 55 (1992), pp. 109–124.

OPTIMUM PROBLEMS WITH CERTAIN LOWER SEMICONTINUOUS SET-VALUED CONSTRAINTS*

ZSOLT PÁLES[†] AND VERA ZEIDAN[‡]

Abstract. In this paper, we provide a complete analysis of second-order admissible variations to inequality type constraints that are given in terms of lower semicontinuous set-valued functions whose images are closed convex sets with nonempty interior. As an application, we consider optimization problems where such constraints are present and we deduce second-order necessary conditions for optimality.

Key words. set-valued maps, lower semicontinuous, nonempty interior, convex sets in $C(T, \mathbf{R}^r)$, support functional, tangent and normal cones, second-order admissible variations

AMS subject classifications. Primary 90C48, 90C34; Secondary 58C06, 47H04

PII. S105262349630725X

1. Introduction. Consider the following optimization problem:

$$(\mathcal{P}) \quad \text{Minimize } f(z) \quad \text{subject to } G(z) \in \mathbf{Q}, H(z) = 0,$$

where $f : D \rightarrow \mathbf{R}$, $G : D \rightarrow X$, $H : D \rightarrow Y$, X, Y, Z are Banach spaces, $D \subset Z$ is nonempty and open, and $\mathbf{Q} \subset X$ is a closed convex set with nonempty interior.

The prototype of such problems arises, for instance, in optimal control theory with state constraints in the inclusion form $x(t) \in Q(t)$ ($\forall t \in T$), where $Q(\cdot)$ is a lower semicontinuous set-valued map with closed convex nonempty interior images.

Better understanding of optimality conditions is an ongoing research program for several researchers. This question is of great value in theory and in applications. As is widely known, such conditions must always be given in terms of the original data of the problem and, in the context of necessity, are expected to be as strong as they can be.

In 1988, Kawasaki [6], [8] discovered, for the problem $(\mathcal{P})$, where $\mathbf{Q}$ is a cone, second-order necessary conditions that contain an extra term manifesting the presence of infinitely many inequalities in the constraint $G(z) \in \mathbf{Q}$. This phenomenon is known as an “envelope-like effect.” Such a result was generalized by Cominetti in [2]. Both results assumed a Mangasarian–Fromovitz-type condition and evoked in the extra term the second-order tangent set:

$$T^2(x, d|\mathbf{Q}) = \liminf_{t \rightarrow 0+} \frac{\mathbf{Q} - x - td}{t^2}.$$

In [11] the authors have generalized the previous results in [6], [8], and [2] to the nondifferentiable case without assuming a Mangasarian–Fromowitz condition. However, instead of $T^2(x, d|\mathbf{Q})$, they have used the second-order admissible variation set

* Received by the editors July 24, 1996; accepted for publication (in revised form) July 2, 1997; published electronically June 3, 1998.

<http://www.siam.org/journals/siopt/8-3/30725.html>

[†] Institute of Mathematics, L. Kossuth University, H-4010 Debrecen, Pf. 12, Hungary (pales@borel.math.klte.hu). The research of this author was supported by Hungarian National Foundation for Scientific Research (OTKA) grant T-016846.

[‡] Department of Mathematics, Michigan State University, East Lansing, MI 48824 (zeidan@math.msu.edu). The research of this author was supported by the National Science Foundation under grant DMS-9404591.

$V(x, d|\mathbf{Q})$ whose elements are vectors $v \in X$ satisfying the following condition: there exists $\bar{\varepsilon} > 0$ such that

$$x + \varepsilon d + \varepsilon^2(v + w) \in \mathbf{Q} \quad \text{for all } 0 < \varepsilon < \bar{\varepsilon}, \|w\| < \bar{\varepsilon}, w \in X.$$

That is,

$$V(x, d|\mathbf{Q}) = \bigcup_{\bar{\varepsilon} > 0} \bigcap_{\substack{\varepsilon < \bar{\varepsilon} \\ \|w\| < \bar{\varepsilon}}} \left[\frac{1}{\varepsilon^2}(\mathbf{Q} - x - \varepsilon d) + w \right].$$

It follows directly from the definition that $V(x, d|\mathbf{Q})$ is an open convex set. Moreover, it is shown in [11] that

$$\overline{V(x, d|\mathbf{Q})} = T^2(x, d|\mathbf{Q}).$$

The results obtained in [11] are analogous to those obtained by Maruyama [9, Theorem 3.2], who uses a Neustadt derivative to handle the nonsmoothness of data.

In order to recall the first- and second-order necessary conditions for $(\mathcal{P})$, obtained in [11, Corollary 2], we need to introduce the following notation and notions.

A point $\hat{z} \in D$ is called an *admissible point* for $(\mathcal{P})$ if $G(\hat{z}) \in \mathbf{Q}$ and $H(\hat{z}) = 0$ hold.

A point $\hat{z} \in D$ is called a *regular point* for $(\mathcal{P})$ if

- (R1) f is locally Lipschitz at $\hat{z}$;
- (R2) G is strictly Fréchet differentiable at $\hat{z}$;
- (R3) H is strictly Fréchet differentiable at $\hat{z}$ and the range of the linear operator $H'(\hat{z})$ is a closed subspace of Y .

If f is locally Lipschitz at $\hat{z}$, then the expression

$$f^o(\hat{z}; d) := \limsup_{(z, \varepsilon) \rightarrow (\hat{z}, 0+)} \frac{f(z + \varepsilon d) - f(z)}{\varepsilon}$$

is finite and will be called *Clarke's generalized directional derivative*. For properties, see [1].

Let $\hat{z}$ be an admissible regular point for $(\mathcal{P})$ and $d \in Z$. A vector $d \in Z$ is called a *critical direction* at $\hat{z}$ for $(\mathcal{P})$ if

- (C1) $f^o(\hat{z}; d) \leq 0$;
- (C2) $G'(\hat{z})d \in \overline{\text{cone}}(\mathbf{Q} - G(\hat{z})) =: T(G(\hat{z})|\mathbf{Q})$;
- (C3) $H'(\hat{z})d = 0$.

A vector $d \in Z$ is called a *regular direction* at $\hat{z}$ for $(\mathcal{P})$ if the following hold.

- (R4) $f^{oo}(\hat{z}, d) := \limsup_{\varepsilon \rightarrow 0+} 2 \frac{f(\hat{z} + \varepsilon d) - f(\hat{z}) - \varepsilon f^o(\hat{z}; d)}{\varepsilon^2}$ is finite.
- (R5) The second-order directional derivatives

$$G''(\hat{z}, d) := \lim_{\varepsilon \rightarrow 0+} 2 \frac{G(\hat{z} + \varepsilon d) - G(\hat{z}) - \varepsilon G'(\hat{z})d}{\varepsilon^2}$$

and

$$H''(\hat{z}, d) := \lim_{\varepsilon \rightarrow 0+} 2 \frac{H(\hat{z} + \varepsilon d) - H(\hat{z}) - \varepsilon H'(\hat{z})d}{\varepsilon^2}$$

exist.

(R6) The set of *second-order admissible variations* of $\mathbf{Q}$ at $G(\hat{z})$ in the direction $G'(\hat{z})d$ is not empty, i.e. $V(G(\hat{z}), G'(\hat{z})d|\mathbf{Q}) \neq \emptyset$.

Clearly, the zero vector is always a regular critical direction at $\hat{z}$ for $(\mathcal{P})$.

Now we are ready to state the result of [11, Corollary 2].

THEOREM 1.1. *Let $\hat{z}$ be a regular solution of the above problem $(\mathcal{P})$. Then, for all regular critical directions d , there exist Lagrange multipliers $\lambda \geq 0$, $x^* \in X^*$, and $y^* \in Y^*$ such that at least one of them is different from zero and the following relations hold.*

$$(1.1) \quad x^* \in N(G(\hat{z})|\mathbf{Q}),$$

$$(1.2) \quad \lambda f''(\hat{z}; z) + \langle x^*, G'(\hat{z})z \rangle + \langle y^*, H'(\hat{z})z \rangle \geq 0 \quad \text{for } z \in Z,$$

and

$$(1.3) \quad \lambda f''''(\hat{z}, d) + \langle x^*, G''(\hat{z}, d) \rangle + \langle y^*, H''(\hat{z}, d) \rangle \geq 2\delta^*(x^*|V(G(\hat{z}), G'(\hat{z})d|\mathbf{Q})).$$

(Here δ^* stands for the support function [14].)

Results along the line of Theorem 1.1 were obtained by Ioffe [5] and Penot [13] for the differentiable case and in the presence of a certain qualification condition. In these works, the superior second-order tangent set is used, namely,

$$\mathbf{Q}''(x, d) := \limsup_{t \rightarrow 0+} \frac{\mathbf{Q} - x - td}{t^2}.$$

Clearly, $\mathbf{Q}''(x, d)$ is larger than $T^2(x, d|\mathbf{Q}) = \overline{V(x, d|\mathbf{Q})}$, but as noted in [7], this set is more difficult to compute and hence is not as useful in the applications.

The relationship between V and T^2 shows that either of the two sets can be used in Theorem 1.1. Two important questions naturally arise from Theorem 1.1.

- (i) How can we check the nonemptiness of $V(x, d|\mathbf{Q})$, since otherwise the second-order optimality conditions would be satisfied trivially?
- (ii) How do we evaluate the support function at $V(x, d|\mathbf{Q})$ and hence $V(x, d|\mathbf{Q})$?

In order that $V(x, d|\mathbf{Q})$ be nonempty, it is only necessary that $\mathbf{Q}$ have a nonempty interior and that d belong to $\overline{\text{cone}(\mathbf{Q} - x)} = T(x|\mathbf{Q})$, which is the tangent cone to $\mathbf{Q}$ at x . If $d \in \text{cone}(\mathbf{Q} - x)$, then $V(x, d|\mathbf{Q})$ is nonempty and $V(x, d|\mathbf{Q}) = \text{cone}(\text{cone}(\text{int } \mathbf{Q} - x) - d)$ (cf. [11, Theorem 4]). In this case the extra term in the second-order condition (1.3) disappears. However, examples are provided by Kawasaki [6] in order to show that the necessary conditions with extra term, that is, when $d \in \overline{\text{cone}(\mathbf{Q} - x)}$, handle situations that cannot be handled with previous results where d is taken from $\text{cone}(\mathbf{Q} - x)$. Thus one has to also consider directions $d \in T(x|\mathbf{Q}) \setminus \text{cone}(\mathbf{Q} - x)$. In this case the description of $V(x, d|\mathbf{Q})$ and its nonemptiness is far from being trivial.

The key role of the two questions stated above is also stressed by Kawasaki [7], to which Penot [13] also refers the readers. In the same paper [7], Kawasaki was able to answer these two questions pertaining to the set $T^2(x, d|\mathbf{Q})$ for only the special case when $\mathbf{Q}$ has the form

$$\mathbf{Q} := C_+(T) := \{ x \in C(T, \mathbf{R}) : x(t) \geq 0 \ \forall t \in T \},$$

where T is a compact metric space. These results, pertaining to the set V , were generalized in [11, Theorem 3] by the authors for the nondifferentiable setting.

In the following theorem we recall the results of Kawasaki [7, Theorem 2.1], [8, Lemma 3.1].

THEOREM 1.2. *Let $x \in C_+(T)$ and $d \in \overline{\text{cone}}(C_+(T) - x)$. Let T_0 be the set of $t_0 \in T$ satisfying the following property: there exists a sequence $t_n \in T$ such that*

$$(1.4) \quad x(t_n) > 0, \quad t_n \rightarrow t_0, \quad \text{and} \quad \frac{d(t_n)}{x(t_n)} \rightarrow -\infty \quad \text{as } n \rightarrow \infty.$$

Define the function $\mathbf{E} = \mathbf{E}_{x,d}$ by

$$\mathbf{E}(t) := \begin{cases} \sup \left\{ \limsup_{n \rightarrow \infty} \frac{d(t_n)^2}{4x(t_n)} : (t_n) \text{ satisfies (1.4)} \right\} & \text{if } t \in T_0, \\ 0 & \text{if } x(t) = d(t) = 0, t \notin T_0, \\ -\infty & \text{otherwise.} \end{cases}$$

Then

- (a) $T^2(x, d|\mathbf{Q}) \neq \emptyset$ if and only if $\mathbf{E}_{x,d}(t) < +\infty$ for all $t \in T$;
- (b) a function $v \in C(T, \mathbf{R})$ belongs to $T^2(x, d|C_+(T))$ if and only if $\mathbf{E}_{x,d}(t) \leq v(t)$ for all $t \in T$.

The function $\mathbf{E}$ defined in Theorem 1.2 can be expressed in another form; see Remark 3.3 below.

Of much more interest is the case when $\mathbf{Q}$ is a subset of $C(T, \mathbf{R}^r)$ defined by

$$(1.5) \quad \mathbf{Q} = \{ x(\cdot) \in C(T, \mathbf{R}^r) \mid x(t) \in Q(t) \ \forall t \in T \},$$

where $Q(\cdot)$ is a lower semicontinuous set-valued map whose images are closed, convex, and have nonempty interior. The importance of this type of constraint stems from control problems with state constraints. Indeed, it is stated by Loewen and Rockafellar in [10]: “A satisfactory treatment of problems whose velocity constraint is described intrinsically by a differential inclusion is one of the principal accomplishments of nonsmooth analysis.”

For the important case, when $\mathbf{Q}$ is given by (1.5), the two questions (i) and (ii) stated above are still open. In this case, the above question can be rephrased as follows:

- (*) Characterize the nonemptiness and the support functional of $V(x, d|\mathbf{Q})$ in terms of the images $Q(t)$ and their support functionals $\delta^*(\cdot|Q(t))$.

Note that, by [12], the set $\mathbf{Q}$ defined by (1.5) is “ C -convex” (see section 2) and thus $V(x, d|\mathbf{Q})$ is also “ C -convex.” Therefore, $V(x, d|\mathbf{Q})$ can be identified with a lower semicontinuous set-valued function $V(x, d|\mathbf{Q})(\cdot)$ whose images are nonempty convex open sets. In this case, by [15] and Theorem 2.1, $\delta^*(\cdot|V(x, d|\mathbf{Q}))$ is expressed in terms of $\delta^*(\cdot|V(x, d|\mathbf{Q})(\cdot))$ and, by [12] and Corollary 2.3, the nonemptiness of $V(x, d|\mathbf{Q})$ is characterized by

$$(1.6) \quad \delta^* \left(\xi | V(x, d|\mathbf{Q})(t) \right) > -\infty \quad \forall (t, \xi) \in T \times \mathbf{R}^r.$$

Furthermore, the correspondence $V(x, d|\mathbf{Q}) \rightarrow q(\cdot, \cdot)$ is bijective, where

$$(1.7) \quad q(t, \xi) := \delta^* \left(\xi | V(x, d|\mathbf{Q})(t) \right)$$

is jointly lower semicontinuous and sublinear in ξ . Therefore, the questions in (*) are equivalent to finding for $q(\cdot, \cdot)$, defined in (1.7), an expression explicitly given in terms of $\delta^*(\cdot|Q(\cdot))$, $x(\cdot)$, and $d(\cdot)$. This task is far from trivial.

The aim of this paper is to positively answer the open questions in (*) in such a way that extends the result known for the case where $\mathbf{Q} = C_+(T)$. In section 2, we review recent results on convex analysis developed in [12] and needed for the rest of the paper. In section 3, we introduce a function $\sigma(x, d|Q)(t, \xi)$ in terms of $\delta^*(\xi|Q(t))$, $x(\cdot)$, and $d(\cdot)$. This function turns out to be jointly lower semicontinuous and positively homogeneous in ξ . In Theorem 3.2, we provide a characterization of $V(x, d|Q)$ in terms of $\overline{\text{co}}\sigma$, which is the largest function below σ that is lower semicontinuous in (t, ξ) and sublinear in ξ . In Theorem 3.5, we show that the nonemptiness of $V(x, d|Q)$ is equivalent to the boundedness below of either the values of $\sigma(x, d|Q)$ or of the values of $\overline{\text{co}}\sigma(x, d|Q)$. Combining these two theorems, we deduce that $\overline{\text{co}}\sigma(x, d|Q)$, which is in terms of $\delta^*(\cdot|Q(\cdot))$, is the function $q(\cdot, \cdot)$, defined in (1.7). Therefore, a thorough answer to (*) is obtained. Furthermore, we show in Corollary 3.9 that not only $V(x, d|Q)$, but also $\text{cl}V(x, d|Q)$ can be characterized in terms of $\overline{\text{co}}\sigma(x, d|Q)$. In section 4, we show how our results extend the results known in the literature for the special case when $Q(t)$ is the cone of nonnegative numbers. Specifically, a comparison with the relevant results of [6] and [7] is displayed. In section 5, we apply the results of section 3 to an optimization problem, which was the initial motivation for those results. There, Theorem 5.1 presents second-order optimality conditions in terms of the new function $\overline{\text{co}}\sigma$. At the end of this section, we also present an example illustrating how the function $\overline{\text{co}}\sigma$ can be evaluated.

2. Preliminary results from convex analysis. Let T be a compact Hausdorff space and $\Gamma(T)$ be the set of lower semicontinuous set-valued maps $Q(\cdot)$ from T to the subsets of $\mathbf{R}^r$ such that $Q(t) \neq \emptyset$ is closed and convex for all $t \in T$. Consider $\Lambda(T)$ the set of jointly lower semicontinuous functions $q : T \times \mathbf{R}^r \rightarrow]-\infty, +\infty]$ such that $q(t, \cdot)$ is sublinear on $\mathbf{R}^r$ for all $t \in T$. Set $\Sigma(T)$ to be the set of nonempty, closed, and C -convex subsets $\mathbf{Q}$ of $C(T, \mathbf{R}^r)$, where the concept of C -convexity is defined as

$$\lambda(\cdot)x(\cdot) + (1 - \lambda(\cdot))y(\cdot) \in \mathbf{Q}$$

whenever $x(\cdot), y(\cdot) \in \mathbf{Q}$, and $\lambda(\cdot) \in C(T, [0, 1])$.

In [12] it is shown that to any $Q(\cdot) \in \Gamma(T)$ there corresponds a function $q(\cdot, \cdot) \in \Lambda(T)$ and a set $\mathbf{Q} \in \Sigma(T)$ as follows:

$$(2.1) \quad q(t, \xi) := \delta^*(\xi|Q(t)) \quad \text{on } T \times \mathbf{R}^r,$$

where δ^* is the support functional and

$$(2.2) \quad \mathbf{Q} := \{ x(\cdot) \in C(T, \mathbf{R}^r) \mid x(t) \in Q(t) \ \forall t \in T \}.$$

Furthermore, the above correspondences between the three sets are bijective and their inverses, respectively, are given by the following formulae:

$$(2.3) \quad Q(t) = \{ x \in \mathbf{R}^r \mid q(t, \xi) \geq \langle \xi, x \rangle \ \forall \xi \in \mathbf{R}^r \}$$

and

$$(2.4) \quad Q(t) = \{ x(t) \mid x(\cdot) \in \mathbf{Q} \}.$$

(Cf. [12, Theorems 4.1 and 4.2].)

Since $\mathbf{Q}$ defined by (2.2) is a convex subset of $C(T, \mathbf{R}^r)$, then its support functional is defined by

$$(2.5) \quad \delta^*(\mu|\mathbf{Q}) := \sup \left\{ \int_T \langle x(t), d\mu(t) \rangle \mid x(\cdot) \in \mathbf{Q} \right\},$$

where μ is a Radon measure on T with values in $\mathbf{R}^r$.

The following connections between the support functionals $\delta^*(\cdot|\mathbf{Q})$ and $\delta^*(\cdot|Q(t))$, the normal cones $N(\cdot|\mathbf{Q})$ and $N(\cdot|Q(t))$, and the tangent cones $T(\cdot|\mathbf{Q})$ and $T(\cdot|Q(t))$ are derived in [16] and [12].

Let $\mu \in (C(T, \mathbf{R}^r))^*$ be a vector-valued Radon measure on T with components $(\mu_1, \dots, \mu_r)$. Then denote

$$|\mu| := |\mu_1| + \dots + |\mu_r|,$$

where $|\mu_i| = \mu_i^+ + \mu_i^-$, and μ_i^+ and μ_i^- are, respectively, the positive and negative parts in the Jordan decomposition of the measure μ_i . It is clear that μ is absolutely continuous with respect to the scalar measure $|\mu|$. We denote by $\frac{d\mu}{d|\mu|}(\cdot)$ the Radon–Nikodym derivative of μ with respect to $|\mu|$.

THEOREM 2.1 (see [12], [16]). *Let $Q(\cdot) \in \Gamma(T)$ and let $\mathbf{Q}$ be defined by (2.2). Then, for $\mu \in (C(T, \mathbf{R}^r))^*$ and $x(\cdot) \in \mathbf{Q}$,*

- (i) $\delta^*(\mu|\mathbf{Q}) = \int_T \delta^*(\frac{d\mu}{d|\mu|}(t)|Q(t))d|\mu|(t)$,
- (ii) $\mu \in N(x(\cdot)|\mathbf{Q})$ iff $\frac{d\mu}{d|\mu|}(t) \in N(x(t)|Q(t))$ $|\mu|$ -a.e.,
- (iii) for $u(\cdot) \in C(T, \mathbf{R}^r)$,

$$u(\cdot) \in T(x(\cdot)|\mathbf{Q}) \quad \text{iff} \quad u(t) \in T(x(t)|Q(t)) \quad \forall t \in T.$$

Now consider the following subsets of $\Gamma(T)$, $\Lambda(T)$, and $\Sigma(T)$:

$$\begin{aligned} \Gamma_0(T) &:= \{ Q(\cdot) \in \Gamma(T) \mid \text{int } Q(t) \neq \emptyset \ \forall t \in T \}, \\ \Lambda_0(T) &:= \{ q \in \Lambda(T) \mid q(t, \xi) + q(t, -\xi) > 0 \ \forall \xi \neq 0, \forall t \in T \}, \\ \Sigma_0(T) &:= \{ \mathbf{Q} \in \Sigma(T) \mid \text{int } \mathbf{Q} \neq \emptyset \}. \end{aligned}$$

The following result was developed in [12]. It states that the three subsets defined above are also equivalent. Furthermore, the connections between the interiors of the sets $\mathbf{Q}$ and $Q(t)$ and the set

$$\{ x \in \mathbf{R}^r \mid q(t, \xi) > \langle \xi, x \rangle \ \forall \xi \neq 0 \}$$

are presented.

THEOREM 2.2 (see [12, Theorems 4.1 and 4.2]). *The sets $\Gamma_0(T)$, $\Lambda_0(T)$, and $\Sigma_0(T)$ are equivalent via the maps defined in (2.1) and (2.2). Furthermore, for $Q(\cdot) \in \Gamma_0(T)$, we have*

$$\text{int } Q(t) = \{ x \in \mathbf{R}^r \mid q(t, \xi) > \langle \xi, x \rangle \ \forall \xi \neq 0 \} \quad \forall t \in T$$

and

$$\text{int } \mathbf{Q} = \{ x(\cdot) \in C(T, \mathbf{R}^r) \mid x(t) \in \text{int } Q(t) \ \forall t \in T \} \neq \emptyset,$$

where q and $\mathbf{Q}$ are defined in (2.1) and (2.2), respectively.

Using the results of Theorem 2.2, the relationship between nonempty open C -convex subsets of $C(T, \mathbf{R}^r)$, lower semicontinuous set-valued maps with nonempty open and convex images, and functionals in $\Lambda_0(T)$ can also be established. Let

$$\Gamma_0(T) := \{ Q(\cdot) : T \rightarrow 2^{\mathbf{R}^r} \mid Q(\cdot) \text{ is lsc and } Q(t) \neq \emptyset \text{ is open and convex } \forall t \in T \},$$

$$\Sigma_0(T) := \{ \mathbf{Q} \subset C(T, \mathbf{R}^r) \mid \mathbf{Q} \neq \emptyset \text{ is open and } C\text{-convex} \}.$$

COROLLARY 2.3 (see [12, Corollary 4.4]). *The set $\mathbf{\Gamma}_0(T)$ is equivalent to $\Lambda_0(T)$ and $\Sigma_0(T)$ via the correspondences defined by (2.1) and (2.2), respectively.*

REMARK 2.1. Let X be a Banach space and let $\mathbf{Q} \subset X$ be a convex set with nonempty interior and $x \in \mathbf{Q}$. Then a vector $d \in X$ is called a *first-order admissible variation of $\mathbf{Q}$ at x* if there exists $\bar{\varepsilon} > 0$ such that

$$x + \varepsilon(d + w) \in \mathbf{Q} \quad \text{for all } 0 < \varepsilon < \bar{\varepsilon}, \|w\| < \bar{\varepsilon}, w \in X.$$

The set of first-order admissible variations is the Dubovitskii–Milyutin cone defined in [3]. A thorough investigation of this cone is important in obtaining first-order necessary conditions concerning inequality constraints (see [4]). It is also known (cf. [4, Theorem 8.2]) that this cone is given by

$$\{ \lambda(u - x) \mid u \in \text{int } \mathbf{Q}, \lambda > 0 \}.$$

It is easy to see that the closure of this set is equal to the cone $T(x|\mathbf{Q})$. Thus the set of first-order admissible variations can also be represented as the interior of $T(x|\mathbf{Q})$.

Consider the case $X = C(T, \mathbf{R}^r)$ and $\mathbf{Q}$ given by (2.2) for $Q(\cdot)$ in $\Gamma(T)$. Thus, by combining Theorems 2.1 and 2.2, we obtain that $d(\cdot)$ is an admissible first-order variation with respect to the set $\mathbf{Q}$ at $x(\cdot) \in \mathbf{Q}$ if and only if

$$d(t) \in \text{int } T(x(t)|Q(t)) \quad \text{for } t \in T.$$

This represents a characterization of the set of first-order variations of $\mathbf{Q}$ in terms of $Q(\cdot)$.

3. Second-order admissible variations. Let X be a Banach space and let $\mathbf{Q} \subset X$ be a closed convex set. In addition, let $x \in \mathbf{Q}$ and $d \in X$. As defined in section 1, a vector $v \in X$ is called a *second-order admissible variation of $\mathbf{Q}$ at x in the direction d* if there exists $\bar{\varepsilon} > 0$ such that

$$x + \varepsilon d + \varepsilon^2(v + w) \in \mathbf{Q} \quad \text{for all } 0 < \varepsilon < \bar{\varepsilon}, \|w\| < \bar{\varepsilon}, w \in X.$$

In the sequel, set $X = C(T, \mathbf{R}^r)$, $T = (T, \rho)$ to be a compact metric space, and set $Q : T \rightarrow 2^{\mathbf{R}^r}$ to be a lower semicontinuous set-valued function whose images are closed and convex with nonempty interior. Define the set $\mathbf{Q} \subset C(T, \mathbf{R}^r)$ by

$$(3.1) \quad \mathbf{Q} = \{ x(\cdot) \in C(T, \mathbf{R}^r) \mid x(t) \in Q(t) \}.$$

Then $\text{int } \mathbf{Q} \neq \emptyset$ by Theorem 2.2.

The main results of this paper are given in this section. We focus on characterizing the set $V(x, d|\mathbf{Q})$ in terms of the images $Q(t)$ and their support functionals. As we shall see this is obtained through the function σ .

Let $x \in \mathbf{Q}$ and $d \in T(x|\mathbf{Q})$ be arbitrarily fixed continuous functions. In order to describe $V(x, d|\mathbf{Q})$, we have to introduce now the following notation.

Let

$$\mathcal{C}(x, d|Q) := \{ (t, \xi) \in T \times \mathbf{R}^r \mid \xi \in N(x(t)|Q(t)), \langle \xi, d(t) \rangle = 0 \}$$

and

$$\mathcal{C}(x, d|Q)(t) := \{ \xi \in \mathbf{R}^r \mid (t, \xi) \in \mathcal{C}(x, d|Q) \}.$$

LEMMA 3.1. *The set-valued map $\mathcal{C}(x, d|Q)(\cdot)$ is nonempty closed convex cone valued; furthermore, $\mathcal{C}(x, d|Q)$ is a closed set.*

Proof. We have

$$(3.2) \quad \mathcal{C}(x, d|Q)(t) = N(x(t)|Q(t)) \cap \{d(t)\}^\perp.$$

Hence $\mathcal{C}(x, d|Q)(t)$ is a nonempty closed convex cone. It remains to prove only the closedness of $\mathcal{C}(x, d|Q)$. Let $t_n \rightarrow t_0$ and $\xi_n \in \mathcal{C}(x, d|Q)(t_n)$ with $\xi_n \rightarrow \xi_0$ be arbitrary sequences. Then, by (3.2),

$$\langle \xi_n, x(t_n) \rangle = \delta^*(\xi_n|Q(t_n)) \quad \langle \xi_n, d(t_n) \rangle = 0.$$

Using the lower semicontinuity of $\delta^*(\cdot|Q(\cdot))$ (see [12, Lemma 3.3]) and upon taking the liminf, we obtain

$$\langle \xi_0, x(t_0) \rangle \geq \delta^*(\xi_0|Q(t_0)) \quad \text{and} \quad \langle \xi_0, d(t_0) \rangle = 0.$$

The first inequality can only be equality since $x(t_0) \in Q(t_0)$ and $\langle \xi_0, x(t_0) \rangle \leq \delta^*(\xi_0|Q(t_0))$ is always true. Thus $\xi_0 \in \mathcal{C}(x, d|Q)(t_0)$, from which the closedness of $\mathcal{C}(x, d|Q)$ follows. $\square$

Let

$$\mathcal{S}(x, d|Q) := \{ (t, \xi) \in T \times \mathbf{R}^r \mid \delta^*(\xi|Q(t)) > \langle \xi, x(t) \rangle, \langle \xi, d(t) \rangle > 0 \},$$

and denote by $\mathcal{S}'(x, d|Q)$ the set of accumulation points of $\mathcal{S}(x, d|Q)$; i.e.,

$$\mathcal{S}'(x, d|Q) = \{ (t, \xi) \mid \exists t_n \rightarrow t, \exists \xi_n \rightarrow \xi \text{ with } (t_n, \xi_n) \in \mathcal{S}(x, d|Q) \forall n \}.$$

Then, obviously, $\mathcal{S}(x, d|Q)$ is an open subset of $T \times \mathbf{R}^r$. Define

$$(3.3) \quad \sigma(x, d|Q)(t, \xi) := \begin{cases} \liminf_{(s, \zeta) \rightarrow (t, \xi)} E(x, d|Q)(s, \zeta), & \text{if } (t, \xi) \in \mathcal{C}(x, d|Q) \cap \mathcal{S}'(x, d|Q), \\ 0 & \text{if } (t, \xi) \in \mathcal{C}(x, d|Q) \setminus \mathcal{S}'(x, d|Q), \\ +\infty & \text{otherwise,} \end{cases}$$

where

$$(3.4) \quad E(x, d|Q)(t, \xi) := \frac{[\langle \xi, d(t) \rangle]^2}{4[\langle \xi, x(t) \rangle - \delta^*(\xi|Q(t))]}.$$

REMARK 3.1. First, note that the ratio $E(x, d|Q)(t, \xi)$ is positively homogeneous in ξ ; therefore, the liminf in the definition of the function $\sigma(x, d|Q)$ could be restricted to those ζ with $|\zeta| = |\xi|$ if $\xi \neq 0$. Furthermore, $\mathcal{C}(x, d|Q)(t)$ and $\mathcal{S}(x, d|Q)(t)$, which is defined similarly to $\mathcal{C}(x, d|Q)(t)$, are cones; thus it follows that $\sigma(x, d|Q)(t, \cdot)$ is a positively homogeneous function. As we shall see in Lemma 3.3 below, the function $\sigma(x, d|Q)(\cdot, \cdot)$ is in fact lower semicontinuous on $T \times (\mathbf{R}^r \setminus \{0\})$.

Define the convex regularization of $\sigma(x, d|Q)(t, \cdot)$ by

$$(3.5) \quad \begin{aligned} \text{co } \sigma(x, d|Q)(t, \xi) &:= \inf \left\{ \sum_{i=1}^n \lambda_i \sigma(x, d|Q)(t, \xi_i) \mid \right. \\ &\quad \left. \xi = \sum_{i=1}^n \lambda_i \xi_i, \xi_i \in \mathbf{R}^r, \lambda_i \in [0, 1], \sum_{i=1}^n \lambda_i = 1 \right\}. \end{aligned}$$

Hence $\text{co } \sigma$ is the largest convex function in ξ such that

$$\text{co } \sigma(x, d|Q)(t, \xi) \leq \sigma(x, d|Q)(t, \xi) \quad \text{for } t \in T, \xi \in \mathbf{R}^r.$$

Consequently, $\text{co } \sigma(x, d|Q)(t, \cdot)$ is sublinear for each fixed $t \in T$.

Now consider the lower semicontinuous regularization in (t, ξ) of $\text{co } \sigma$. It is denoted by $\overline{\text{co}} \sigma$ and is defined as

$$(3.6) \quad \overline{\text{co}} \sigma(x, d|Q)(t, \xi) := \liminf_{(\tau, \zeta) \rightarrow (t, \xi)} \text{co } \sigma(x, d|Q)(\tau, \zeta).$$

Clearly, $\overline{\text{co}} \sigma$ is the largest lower semicontinuous function in (t, ξ) that is bounded above by $\text{co } \sigma$. The result is that $\overline{\text{co}} \sigma(x, d|Q)(t, \cdot)$ is also sublinear. Therefore, $\overline{\text{co}} \sigma$ is in $\Lambda(T)$.

The aim of this section is twofold:

- (i) to establish a characterization of the set $V(x, d|Q)$ in terms of the convex and jointly lower semicontinuous function $\overline{\text{co}} \sigma(x, d|Q)$ (see Theorem 3.2);
- (ii) to provide a full description of the nonemptiness of $V(x, d|Q)$ in terms of the function $\sigma(x, d|Q)$ or $\overline{\text{co}} \sigma(x, d|Q)$ (see Theorem 3.5).

The combination of these results confirms that $\overline{\text{co}} \sigma$ is the function q in (1.7) corresponding to the set-valued map $V(x, d|Q)(\cdot)$. Another consequence is that $\overline{\text{co}} \sigma$ is also the function q in (2.1) corresponding to the set-valued map $\bar{V}(x, d|Q)(\cdot)$, where

$$(3.7) \quad \bar{V}(x, d|Q)(t) := \{ v(t) \in \mathbf{R}^r \mid v(\cdot) \in \text{cl } V(x, d|Q) \}$$

(see Corollary 3.9).

THEOREM 3.2. *A function $v : T \rightarrow \mathbf{R}^r$ belongs to $V(x, d|Q)$ if and only if*

$$(3.8) \quad \langle \xi, v(t) \rangle < \overline{\text{co}} \sigma(x, d|Q)(t, \xi) \quad \text{for all } t \in T, \xi \in (\mathbf{R}^r \setminus \{0\}).$$

The proof is based on the following two results. The first is a modification of the key lemma from [11].

LEMMA 3.3. *Let $\mathcal{T}$ be a compact metric space and let $a : \mathcal{T} \rightarrow [-\infty, \infty[$ and $b, c : \mathcal{T} \rightarrow \mathbf{R}$ be upper semicontinuous functions. Define $\sigma_{a,b} : \mathcal{T} \rightarrow [-\infty, \infty]$ by*

$$\sigma_{a,b}(u) := \begin{cases} \liminf_{\substack{\tau \rightarrow u \\ a(\tau) < 0, b(\tau) > 0}} \frac{b^2(\tau)}{4a(\tau)} & \text{if } u \in \mathcal{T}_{a=0, b=0} \cap \partial(\mathcal{T}_{a<0, b>0}), \\ 0 & \text{if } u \in \mathcal{T}_{a=0, b=0} \setminus \partial(\mathcal{T}_{a<0, b>0}), \\ +\infty & \text{otherwise.} \end{cases}$$

Then $\sigma_{a,b}$ is a lower semicontinuous function and the set of points where $\sigma_{a,b} < 0$ is nowhere dense in $\mathcal{T}$. Furthermore, the following statements are equivalent to each other

(i)

$$\begin{aligned} a(u) &\leq 0 && \text{for all } u \in \mathcal{T}, \\ b(u) &\leq 0 && \text{for all } u \in \mathcal{T}_{a=0}, \\ c(u) &< \sigma_{a,b}(u) && \text{for all } u \in \mathcal{T}. \end{aligned}$$

(ii) *There exists $\delta > 0$ such that, for all $u_0 \in \mathcal{T}$, for all sequences $u_n \in \mathcal{T}$ with $u_n \rightarrow u_0$ and $\varepsilon_n > 0$ with $\varepsilon_n \rightarrow 0$,*

$$a(u_n) + \varepsilon_n b(u_n) + \varepsilon_n^2 (c(u_0) + \delta) \leq 0$$

holds for sufficiently large values of $n \in \mathbf{N}$.

If c is also continuous, then the following statement is also equivalent to (i) and (ii).

(iii) There exists $\bar{\varepsilon} > 0$ such that, for all $u \in \mathcal{T}$ and for all $\varepsilon < \bar{\varepsilon}$,

$$a(u) + \varepsilon b(u) + \varepsilon^2(c(u) + \bar{\varepsilon}) \leq 0.$$

REMARK 3.2. The novelty in the formulation of this lemma versus the key lemma in [11] resides in the fact that the functions a and $\sigma_{a,b}$ may take the value $-\infty$ and $+\infty$, respectively, and in the equivalence of the third condition (iii) with each of (i) and (ii).

REMARK 3.3. The function $\sigma_{a,b}$ and the function $\mathbf{E} = \mathbf{E}_{x,d}$ introduced in Theorem 1.2 satisfy the following identity (if x, d are as in Theorem 1.2):

$$\mathbf{E}_{x,d} = -\sigma_{-x,-d}.$$

This relation can be checked immediately. Therefore, the use of the function $\mathbf{E}$ is also possible if a, b are continuous functions.

Proof. The proof of the lower semicontinuity of $\sigma_{a,b}$ and (i) $\iff$ (ii) is analogous to that of the key lemma in [11].

(ii) $\implies$ (iii). Assume that (iii) is not true. Then, for all $\bar{\varepsilon} = 1/n$, there exists $u_n \in \mathcal{T}$, $\varepsilon_n < 1/n$ such that

$$a(u_n) + \varepsilon_n b(u_n) + \varepsilon_n^2 \left(c(u_n) + \frac{1}{n} \right) > 0.$$

We may assume that $u_n \rightarrow u_0$, since $\mathcal{T}$ is compact. By the upper semicontinuity of c , $c(u_n) + 1/n \leq c(u_0) + \delta$ for large n . Thus

$$a(u_n) + \varepsilon_n b(u_n) + \varepsilon_n^2 (c(u_0) + \delta) > 0$$

for large n , which contradicts (ii).

(iii) $\implies$ (ii). Let $\delta := \bar{\varepsilon}/2$ and $u_n \rightarrow u_0$, $\varepsilon_n \rightarrow 0$ be arbitrary sequences. Then, by (iii),

$$a(u_n) + \varepsilon_n b(u_n) + \varepsilon_n^2 (c(u_n) + \bar{\varepsilon}) \leq 0$$

for $\varepsilon_n < \bar{\varepsilon}$, that is, for large values of n . Since c is lower semicontinuous, $c(u_n) > c(u_0) - \delta$ if n is large enough. Thus $c(u_n) + \bar{\varepsilon} > c(u_0) + \delta$ and therefore

$$a(u_n) + \varepsilon_n b(u_n) + \varepsilon_n^2 (c(u_0) + \delta) \leq 0$$

for sufficiently large n . $\square$

The following result is the second key to prove Theorem 3.2. It is a characterization of $V(x, d|\mathbf{Q})$ in terms of σ . Hence it is a weaker result than Theorem 3.2.

LEMMA 3.4. A continuous function $v : T \rightarrow \mathbf{R}^r$ belongs to $V(x, d|\mathbf{Q})$ if and only if

$$\langle \xi, v(t) \rangle < \sigma(x, d|Q)(t, \xi) \quad \text{for all } t \in T, \xi \in \mathbf{R}^r \setminus \{0\}.$$

Proof. A function $v : T \rightarrow \mathbf{R}^r$ belongs to $V(x, d|\mathbf{Q})$ if and only if there exists $\bar{\varepsilon} > 0$ such that

$$x(t) + \varepsilon d(t) + \varepsilon^2(v(t) + w(t)) \in Q(t)$$

for all $t \in T$, $\varepsilon \leq \bar{\varepsilon}$ and continuous functions $w : T \rightarrow \mathbf{R}^r$ with $\|w\| \leq \bar{\varepsilon}$. The set $Q(t)$ is closed and convex; therefore, $p \in Q(t)$ holds if and only if $\langle \xi, p \rangle \leq \delta^*(\xi|Q(t))$ for all $\xi \in \mathbf{R}^r$ with $|\xi| = 1$. Applying this argument, we obtain that $v \in V(x, d|\mathbf{Q})$ if and only if

$$(3.9) \quad \langle \xi, x(t) \rangle + \varepsilon \langle \xi, d(t) \rangle + \varepsilon^2 (\langle \xi, v(t) \rangle + \bar{\varepsilon}) \leq \delta^*(\xi, Q(t))$$

for all $t \in T$, $\varepsilon \leq \bar{\varepsilon}$ and ξ with $|\xi| = 1$.

Take

$$(3.10) \quad \begin{aligned} \mathcal{T} &:= T \times \{ \xi \in \mathbf{R}^r : |\xi| = 1 \}, \\ a(t, \xi) &:= \langle \xi, x(t) \rangle - \delta^*(\xi|Q(t)), \\ b(t, \xi) &:= \langle \xi, d(t) \rangle, \\ c(t, \xi) &:= \langle \xi, v(t) \rangle. \end{aligned}$$

Then $\mathcal{T}$ is a compact metric space, and the functions a, b, c satisfy the conditions of Lemma 3.3. One can also observe that

$$\sigma(x, d|Q)(t, \xi) = \sigma_{a,b}(t, \xi) \quad \text{for } (t, \xi) \in \mathcal{T}.$$

Therefore, by the equivalence of the statements (i) and (iii) of Lemma 3.3, v satisfies (3.9) on the domain indicated if and only if

$$\langle \xi, x(t) \rangle - \delta^*(\xi|Q(t)) \leq 0 \quad \text{for } (t, \xi) \in \mathcal{T},$$

$$\langle \xi, d(t) \rangle \leq 0 \quad \text{for } (t, \xi) \in \mathcal{T} \text{ such that } \langle \xi, x(t) \rangle = \delta^*(\xi|Q(t)),$$

and

$$(3.11) \quad \langle \xi, v(t) \rangle < \sigma(x, d|Q)(t, \xi) \quad \text{for } (t, \xi) \in \mathcal{T}.$$

The first inequality immediately follows from $x(t) \in Q(t)$. The relation $\langle \xi, x(t) \rangle = \delta^*(\xi|Q(t))$ is equivalent to $\xi \in N(x(t)|Q(t))$; therefore, the second inequality is equivalent to $d(t) \in T(x(t)|Q(t))$ which by Theorem 2.1 is the same as $d \in T(x|\mathbf{Q})$. Thus $v \in V(x, d|\mathbf{Q})$ if and only if (3.11) holds. Using the observation on the positive homogeneity in Remark 3.1, we get the statement of this lemma. $\square$

Proof of Theorem 3.2 (Sufficiency). If v satisfies (3.8), then (3.5) and (3.6) yield that v also satisfies the condition of Lemma 3.4; therefore, $v \in V(x, d|\mathbf{Q})$.

Necessity. Assume that $v \in V(x, d|\mathbf{Q})$. Since the set $V(x, d|\mathbf{Q})$ is open, then there exists $\delta > 0$ such that $v + \delta \mathbf{B} \subset V(x, d|\mathbf{Q})$. Thus, by Lemma 3.4,

$$(3.12) \quad \langle \xi, v(t) \rangle + \delta |\xi| < \sigma(x, d|Q)(t, \xi) \quad \text{for all } t \in T, \xi \in (\mathbf{R}^r \setminus \{0\}).$$

By defining $\sigma(x, d|Q)(t, 0) := 0$, the domain of this inequality can be extended to the set $T \times \mathbf{R}^r$.

Let $\xi_1, \dots, \xi_n \in \mathbf{R}^r$ and $\lambda_1, \dots, \lambda_n \in [0, 1]$ with $\lambda_1 + \dots + \lambda_n = 1$. Then it follows from (3.12) that

$$\left\langle \sum_{i=1}^n \lambda_i \xi_i, v(t) \right\rangle + \delta \left| \sum_{i=1}^n \lambda_i \xi_i \right| \leq \sum_{i=1}^n \lambda_i [\langle \xi_i, v(t) \rangle + \delta |\xi_i|] \leq \sum_{i=1}^n \lambda_i \sigma(x, d|Q)(t, \xi_i).$$

Therefore,

$$\begin{aligned} \langle \xi, v(t) \rangle + \delta |\xi| &\leq \text{co } \sigma(x, d|Q)(t, \xi) \\ &= \inf \left\{ \sum_{i=1}^n \lambda_i \sigma(x, d|Q)(t, \xi_i) \mid \xi = \sum_{i=1}^n \lambda_i \xi_i, \xi_i \in \mathbf{R}^r, \lambda_i \in [0, 1], \sum_{i=1}^n \lambda_i = 1 \right\}. \end{aligned}$$

Now fix (t_0, ξ_0) and take the liminf with respect to $(t, \xi) \rightarrow (t_0, \xi_0)$ in the above inequality. We get

$$\langle \xi_0, v(t_0) \rangle + \delta |\xi_0| \leq \liminf_{(t, \xi) \rightarrow (t_0, \xi_0)} \text{co } \sigma(x, d|Q)(t, \xi) = \overline{\text{co}} \sigma(x, d|Q)(t_0, \xi_0).$$

Since (t_0, ξ_0) is arbitrary, (3.8) follows at once. $\square$

The statement below is a characterization of the nonemptiness of $V(x, d|Q)$.

THEOREM 3.5. *The following three statements are equivalent:*

- (i) $V(x, d|Q) \neq \emptyset$,
- (ii) $\overline{\text{co}} \sigma(x, d|Q)(t, \xi) > -\infty$ for all $(t, \xi) \in \mathcal{T}$,
- (iii) $\sigma(x, d|Q)(t, \xi) > -\infty$ for all $(t, \xi) \in \mathcal{T}$.

Proof. (i) $\implies$ (ii). If $v \in V(x, d|Q)$, then by Theorem 3.2,

$$\langle \xi, v(t) \rangle < \overline{\text{co}} \sigma(x, d|Q)(t, \xi) \quad \text{for all } (t, \xi) \in \mathcal{T}.$$

Hence $\overline{\text{co}} \sigma(x, d|Q)(t, \xi) > -\infty$.

The implication (ii) $\implies$ (iii) is obvious. The proof of (iii) $\implies$ (i) is split into three steps.

Step 1. We prove the following lemma.

LEMMA 3.6. *If (iii) holds, then there exists $K > 0$ such that*

$$(3.13) \quad -K|\xi| \leq E(x, d|Q)(t, \xi)$$

if $(t, \xi) \in \mathcal{S}(x, d|Q)$. (Here $E(x, d|Q)(\cdot, \cdot)$ is defined in (3.4).)

Proof. By contradiction assume that for all $n \in \mathbf{N}$ there exists $(t_n, \xi_n) \in \mathcal{S}(x, d|Q)$ with $|\xi_n| = 1$ so that

$$E(x, d|Q)(t_n, \xi_n) < -n.$$

We can assume that $t_n \rightarrow t_0$ and $\xi_n \rightarrow \xi_0$. Observe that the function $E(x, d|Q)(\cdot, \cdot)$ is lower semicontinuous; therefore, we obtain

$$-\infty = \liminf_{n \rightarrow \infty} E(x, d|Q)(t_n, \xi_n) \geq E(x, d|Q)(t_0, \xi_0).$$

That is, $E(x, d|Q)(t_0, \xi_0) = -\infty$. Thus $(t_0, \xi_0) \notin \mathcal{S}(x, d|Q)$. The sequence (t_n, ξ_n) being in this set, (t_0, ξ_0) must belong to the boundary of $\mathcal{S}(x, d|Q)$. This means that at least one of the equalities

$$\delta^*(\xi_0|Q(t_0)) = \langle \xi_0, x(t_0) \rangle, \quad \langle \xi_0, d(t_0) \rangle = 0$$

is valid. The relations

$$\delta^*(\xi_0|Q(t_0)) = \langle \xi_0, x(t_0) \rangle \quad \text{and} \quad \langle \xi_0, d(t_0) \rangle > 0$$

are not possible because $d(t_0)$ is tangent to $Q(t_0)$ at $x(t_0)$. If

$$\delta^*(\xi_0|Q(t_0)) > \langle \xi_0, x(t_0) \rangle \quad \text{and} \quad \langle \xi_0, d(t_0) \rangle = 0,$$

then $E(x, d|Q)(t_0, \xi_0) = 0$, which is a contradiction. The last case is when

$$\delta^*(\xi_0|Q(t_0)) = \langle \xi_0, x(t_0) \rangle \text{ and } \langle \xi_0, d(t_0) \rangle = 0.$$

Then $(t_0, \xi_0) \in \mathcal{C}(x, d|Q)$. Therefore,

$$-\infty = \liminf_{n \rightarrow \infty} E(x, d|Q)(t_n, \xi_n) \geq \sigma(x, d|Q)(t_0, \xi_0) > -\infty,$$

which means a contradiction to (iii). $\square$

Step 2. We express the characterization obtained in Lemma 3.4 in a different form.

LEMMA 3.7. *A function $v : T \rightarrow \mathbf{R}^r$ belongs to $V(x, d|Q)$ if and only if there exists a positive number $\bar{\varepsilon} > 0$ such that*

$$\langle \xi, v(t) \rangle \leq \begin{cases} -\bar{\varepsilon}|\xi| + E(x, d|Q)(t, \xi) & \text{if } \bar{\varepsilon} \langle \xi, d(t) \rangle > 2[\delta^*(\xi|Q(t)) - \langle \xi, x(t) \rangle], \\ -\bar{\varepsilon}|\xi| + \frac{\delta^*(\xi|Q(t)) - \langle \xi, x(t) \rangle - \bar{\varepsilon} \langle \xi, d(t) \rangle}{\bar{\varepsilon}^2} & \text{if } \bar{\varepsilon} \langle \xi, d(t) \rangle \leq 2[\delta^*(\xi|Q(t)) - \langle \xi, x(t) \rangle] \end{cases}$$

for all $t \in T$, $\xi \in (\mathbf{R}^r \setminus \{0\})$.

Proof. As we have already proved in the proof of Lemma 3.4, $v \in V(x, d|Q)$ holds if and only if there exists $\bar{\varepsilon} > 0$ such that (3.9) holds for all $t \in T$, $\varepsilon \leq \bar{\varepsilon}$, and $\xi \in \mathbf{R}^r$ with $|\xi| = 1$. In other words, $v \in V(x, d|Q)$ if and only if there exists $\bar{\varepsilon} > 0$ such that

$$\langle \xi, v(t) \rangle \leq -\bar{\varepsilon}|\xi| + \inf_{0 < \varepsilon \leq \bar{\varepsilon}} \frac{\delta^*(\xi|Q(t)) - \langle \xi, x(t) \rangle - \varepsilon \langle \xi, d(t) \rangle}{\varepsilon^2}$$

for all $t \in T$ and for all $\xi \neq 0$. Computing the infimum on the right-hand side, the result follows immediately. $\square$

Step 3. The following lemma completes the proof of Theorem 3.5.

LEMMA 3.8. *If (iii) holds, then for any $w \in \text{int } Q$ there exists $\bar{\varepsilon}$ so that*

$$v = \frac{w - x - \bar{\varepsilon}d}{\bar{\varepsilon}^2} \in V(x, d|Q).$$

Proof. Let w be in the interior of Q . Then there exists $\delta > 0$ such that $w + \delta \mathbf{B} \subset Q$ (where $\mathbf{B}$ stands for the unit ball of the space $C(T, \mathbf{R}^r)$). Hence

$$\langle \xi, w(t) \rangle + \delta|\xi| \leq \delta^*(\xi|Q(t)) \quad \text{for all } t \in T, \xi \in \mathbf{R}^r.$$

Let K be the constant whose existence was stated in Lemma 3.6. Choose $\bar{\varepsilon} > 0$ so that $\bar{\varepsilon}^2(\bar{\varepsilon} + K) \leq \delta$. Thus we have

$$\langle \xi, w(t) \rangle + \bar{\varepsilon}^2(\bar{\varepsilon} + K)|\xi| \leq \delta^*(\xi|Q(t)) \quad \text{for all } t \in T, \xi \in \mathbf{R}^r.$$

To complete the proof of this lemma, we are going to show that v satisfies the necessary and sufficient condition of Lemma 3.7 with this $\bar{\varepsilon}$.

Substituting $v = (w - x - \bar{\varepsilon}d)/\bar{\varepsilon}^2$ into this condition, it remains to prove that

$$\langle \xi, w(t) \rangle \leq \begin{cases} -\bar{\varepsilon}^3|\xi| + \bar{\varepsilon}^2 E(x, d|Q)(t, \xi) + \langle \xi, x(t) \rangle + \bar{\varepsilon} \langle \xi, d(t) \rangle & \text{if } \bar{\varepsilon} \langle \xi, d(t) \rangle > 2[\delta^*(\xi|Q(t)) - \langle \xi, x(t) \rangle], \\ -\bar{\varepsilon}^3|\xi| + \delta^*(\xi|Q(t)) & \text{if } \bar{\varepsilon} \langle \xi, d(t) \rangle \leq 2[\delta^*(\xi|Q(t)) - \langle \xi, x(t) \rangle] \end{cases}$$

for all $t \in T$, $\xi \in (\mathbf{R}^r \setminus \{0\})$.

By the choice of $\bar{\varepsilon}$,

$$\langle \xi, w(t) \rangle + \bar{\varepsilon}^3 |\xi| \leq \delta^*(\xi|Q(t)) \quad \text{for all } t \in T, \xi \in \mathbf{R}^r.$$

Therefore, it remains to show that the first inequality holds on the domain indicated.

If (t, ξ) belongs to that domain, then

$$\bar{\varepsilon} \langle \xi, d(t) \rangle > 2[\delta^*(\xi|Q(t)) - \langle \xi, x(t) \rangle] \geq \delta^*(\xi|Q(t)) - \langle \xi, x(t) \rangle.$$

The right-hand side of this inequality cannot be zero, otherwise $\langle \xi, d(t) \rangle = 0$, which gives a contradiction. Thus $(t, \xi) \in \mathcal{S}(x, d|Q)$ holds too. Therefore, by Lemma 3.6, (3.13) is valid. Combining these inequalities, we obtain

$$\begin{aligned} \langle \xi, w(t) \rangle &\leq \delta^*(\xi|Q(t)) - \bar{\varepsilon}^2(\bar{\varepsilon} + K)|\xi| \\ &< \bar{\varepsilon} \langle \xi, d(t) \rangle + \langle \xi, x(t) \rangle - \bar{\varepsilon}^3 |\xi| - \bar{\varepsilon}^2 K |\xi| \\ &\leq \bar{\varepsilon} \langle \xi, d(t) \rangle + \langle \xi, x(t) \rangle - \bar{\varepsilon}^3 |\xi| + \bar{\varepsilon}^2 E(x, d|Q)(t, \xi) \end{aligned}$$

for all (t, ξ) satisfying $\bar{\varepsilon} \langle \xi, d(t) \rangle > 2[\delta^*(\xi|Q(t)) - \langle \xi, x(t) \rangle]$. Therefore, it holds that $v \in V(x, d|Q)$. $\square$

Thus the nonemptiness of $V(x, d|Q)$ is also proved and the proof of Theorem 3.5 is complete. $\square$

REMARK 3.4. When nonempty, the set $V(x, d|Q)$ is in $\Sigma_0(T)$. Thus Theorems 3.2 and 3.5 and Corollary 2.3 yield that $\overline{\text{co}} \sigma$ is the function $q(\cdot, \cdot)$ defined by (1.7). Therefore, when one of the equivalent statements of Theorem 3.5 holds, then $\overline{\text{co}} \sigma$ belongs to $\Lambda_0(T)$; that is, in addition to its lower semicontinuity and sublinearity property, it also satisfies

$$\overline{\text{co}} \sigma(x, d|Q)(t, \xi) + \overline{\text{co}} \sigma(x, d|Q)(t, -\xi) > 0$$

for all $t \in T$ and for all $\xi \in \mathbf{R}^r$, $\xi \neq 0$.

The next result is a consequence of Theorems 3.2 and 3.5. It states that the function $q(\cdot, \cdot) = \overline{\text{co}} \sigma(x, d|Q)$, which is in $\Lambda_0(T)$, is the support functional corresponding via the map (2.1) to the set-valued map $\bar{V}(x, d|Q)(\cdot)$, where

$$\bar{V}(x, d|Q)(t) := \{ v(t) \mid v(\cdot) \in \text{cl } V(x, d|Q) \}.$$

COROLLARY 3.9. *The inclusion $v \in \text{cl } V(x, d|Q)$ holds if and only if*

$$(3.14) \quad \langle \xi, v(t) \rangle \leq \overline{\text{co}} \sigma(x, d|Q)(t, \xi) \quad \text{for all } t \in T, \xi \in \mathbf{R}^r.$$

Proof. Let $v \in \text{cl } V(x, d|Q)$. Then there exists a sequence $v_n \in V(x, d|Q)$ such that $v_n \rightarrow v$. Now, by Theorem 3.2,

$$(3.15) \quad \langle \xi, v_n(t) \rangle < \overline{\text{co}} \sigma(x, d|Q)(t, \xi) \quad \text{for all } t \in T, \xi \in (\mathbf{R}^r \setminus \{0\});$$

therefore, by taking the limit $n \rightarrow \infty$, (3.14) follows.

Conversely, if v satisfies (3.14), then $\sigma(x, d|Q)(t, \xi) > -\infty$ for all $t \in T$, $\xi \neq 0$. Therefore, $V(x, d|Q)$ is not empty by Theorem 3.5. Let $v_0 \in V(x, d|Q)$ and let $v_n := (1 - 1/n)v + (1/n)v_0$. Then $v_n \rightarrow v$ and (3.15) holds (by adding the inequality for v and v_0 with the proper convex combination). Thus $v_n \in V(x, d|Q)$; whence $v \in \text{cl } V(x, d|Q)$. $\square$

In the following result, we describe how the support functional of $V(x, d|\mathbf{Q})$ can be evaluated in terms of $\overline{\text{co}} \sigma$.

THEOREM 3.10. *Assume that condition (iii) of Theorem 3.5 holds and let $\mu \in \mathcal{M}(T)$ be a bounded vector-valued Borel measure on T . Then*

$$(3.16) \quad \delta^*(\mu|V(x, d|\mathbf{Q})) = \int_T \overline{\text{co}} \sigma(x, d|Q) \left(t, \frac{d\mu}{d|\mu|}(t) \right) d|\mu|(t),$$

where $\frac{d\mu}{d|\mu|}(\cdot)$ is the Radon–Nikodym derivative of μ with respect to $|\mu|$.

Proof. From the definition of the support functional δ^* , the result is that

$$\delta^*(\mu|V(x, d|\mathbf{Q})) = \delta^*(\mu|\text{cl } V(x, d|\mathbf{Q})).$$

From Corollary 3.9 we have that

$$q(t, \xi) := \overline{\text{co}} \sigma(x, d|Q)(t, \xi)$$

is the support functional on $T \times \mathbf{R}^r$ associated with $\text{cl } V(x, d|\mathbf{Q})$ via the map (2.1). Thus, by Theorem 2.1, (3.16) follows. $\square$

4. Comparison with known results. In this section we specialize the statements of Theorems 3.2, 3.5, and 3.10 to the setting when $Q(\cdot)$ is the set-valued map defined by

$$Q(t) :=] - \infty, 0], \quad t \in T.$$

We show how the function $\overline{\text{co}} \sigma$ relates to the function $\mathbf{E}$ of Theorem 1.2 introduced by Kawasaki. The set $\mathbf{Q}$ defined in (3.1) is now the set of all nonpositive-valued continuous functions. Let $x \in \mathbf{Q}$ and $d \in T(x|\mathbf{Q})$ be fixed. By Theorem 2.1, the latter inclusion is equivalent to the following condition: d is a continuous function satisfying

$$d(t) \leq 0 \quad \text{if} \quad x(t) = 0 \quad (t \in T).$$

LEMMA 4.1. *Under the above assumptions,*

$$(4.1) \quad \overline{\text{co}} \sigma(x, d|Q)(t, \xi) = \begin{cases} \xi \sigma_{x,d}(t) & \text{if } t \in T, \xi > 0, \\ +\infty & \text{if } t \in T, \xi < 0, \end{cases}$$

where $\sigma_{x,d}$ is defined as $\sigma_{a,b}$ in Lemma 3.3 with $a = x$ and $b = d$.

Proof. Taking the definitions used in Lemma 3.4

$$\begin{aligned} a(t, \xi) &:= \xi x(t) - \delta^*(\xi|Q(t)), \\ b(t, \xi) &:= \xi d(t), \end{aligned}$$

we get that

$$\sigma(x, d|Q)(t, \xi) = \sigma_{a,b}(t, \xi) = |\xi| \sigma_{x,d}(t, \text{sgn}(\xi))$$

for $t \in T$ and $\xi \neq 0$. Hence

$$\sigma(x, d|Q)(t, \xi) = \begin{cases} \xi \sigma_{x,d}(t) & \text{if } t \in T, \xi > 0, \\ +\infty & \text{if } t \in T, \xi < 0. \end{cases}$$

The function on the right-hand side of this equation is sublinear in ξ and lower semi-continuous in (t, ξ) on $T \times (\mathbf{R} \setminus \{0\})$. Therefore, $\overline{\text{co}} \sigma(x, d|Q) = \sigma(x, d|Q)$ for $t \in T$ and $\xi \neq 0$. $\square$

COROLLARY 4.2. *Let $\mathbf{Q}$ be the set of all nonpositive-valued continuous functions on the compact metric space T , and let $x \in \mathbf{Q}$ and $d \in T(x|\mathbf{Q})$. Then we have the following assertions:*

- (i) $V(x, d|\mathbf{Q}) \neq \emptyset$ if and only if $\sigma_{x,d}(t) > -\infty$ for all $t \in T$;
- (ii) a continuous function $v : T \rightarrow \mathbf{R}$ belongs to $V(x, d|\mathbf{Q})$ if and only if $v(t) < \sigma_{x,d}(t)$ for all $t \in T$;
- (iii) a continuous function $v : T \rightarrow \mathbf{R}$ belongs to $\text{cl} V(x, d|\mathbf{Q})$ if and only if $v(t) \leq \sigma_{x,d}(t)$ for all $t \in T$;
- (iv) if $V(x, d|\mathbf{Q}) \neq \emptyset$ and $\mu \in \mathcal{M}(T)$ is a nonnegative Borel measure on T , then

$$\delta^*(\mu|V(x, d|\mathbf{Q})) = \int_T \sigma_{x,d}(t) d\mu(t).$$

Proof. Using Lemma 4.1, the statements directly follow from Theorem 3.5, Theorem 3.2, Corollary 3.9, and Theorem 3.10, respectively. $\square$

REMARK 4.1. The first and third statements of this corollary are equivalent to that of Theorem 1.2 (which is the result of Kawasaki [7], [8], since $\text{cl} V(x, d|\mathbf{Q}) = T^2(x, d|\mathbf{Q})$ and $\sigma_{x,d}$ can be replaced by $-\mathbf{E}_{-x,-d}$ as we have noted in Remark 3.3.

REMARK 4.2. If the set-valued map $Q(\cdot)$ is now defined by

$$Q(t) :=]-\infty, 0]^r \subset \mathbf{R}^r,$$

then, for $x = (x_1, \dots, x_r) \in \mathbf{Q}$, $d = (d_1, \dots, d_r) \in T(x|\mathbf{Q})$, the results concerning the set $V(x, d|\mathbf{Q})$ follow from Corollary 4.2 and from the following easy-to-obtain formula

$$V(x, d|\mathbf{Q}) = V(x_1, d_1|\mathbf{Q}_1) \times \dots \times V(x_r, d_r|\mathbf{Q}_r),$$

where $\mathbf{Q}_1 = \dots = \mathbf{Q}_r$ is the set of nonpositive continuous functions.

5. Applications to optimization theory. In this section we make the following specifications of the optimization problem $(\mathcal{P})$ and Theorem 1.1. Let T be a compact Hausdorff space and let $X = C(T, \mathbf{R}^r)$. Furthermore, let $G : D \rightarrow X$ and $\mathbf{Q} \subset X$ be given by

$$(5.1) \quad G(z)(t) = g(t, z) \quad \text{and} \quad \mathbf{Q} := \{ x(\cdot) \mid x(t) \in Q(t) \ \forall t \in T \},$$

where $g : T \times D \rightarrow \mathbf{R}^r$ is a continuous function and $Q : T \rightarrow 2^{\mathbf{R}^r}$ is a lower semicontinuous set-valued map whose values are closed convex sets with nonempty interiors. Then, by Theorem 2.2, the interior of $\mathbf{Q}$ is also nonempty. Thus $(\mathcal{P})$ reduces to the following problem:

$$(\mathcal{P}^*) \quad \text{minimize } f(z) \quad \text{subject to} \quad g(t, z) \in Q(t) \ (\forall t \in T), \quad H(z) = 0.$$

The main focus of this section is to apply Theorem 1.1 to the problem $(\mathcal{P}^*)$ in such a way that all of the hypotheses assumed and all the results obtained will be phrased explicitly in terms of the data $f, g, Q(\cdot)$, and H . In particular, the interpretation of the second-order necessary condition (1.3) is an important task.

- A point $\hat{z} \in D$ is admissible for $(\mathcal{P}^*)$ if $g(t, \hat{z}) \in Q(t)$ for all $t \in T$ and $H(\hat{z}) = 0$.

- The regularity of $\hat{z}$ means that the assumptions (R1) and (R3) are valid and, in addition,
- (R2*) the map $g(t, \cdot)$ is uniformly strictly Fréchet differentiable at $\hat{z}$ for all $t \in T$; that is,

$$\lim_{z_1, z_2 \rightarrow \hat{z}} \frac{|g(t, z_1) - g(t, z_2) - g'(t, \hat{z})(z_1 - z_2)|}{\|z_1 - z_2\|} = 0$$

holds uniformly for $t \in T$.

Then with the definition (5.1) the map G is strictly Fréchet differentiable at $\hat{z}$; therefore, (R2) holds too.

- A vector d is critical for $(\mathcal{P}^*)$ at an admissible regular point $\hat{z}$ if (C1), (C3) are valid; furthermore,
 - (C2*) $g'(t, \hat{z})d \in T(g(t, \hat{z})|Q(t))$ for all $t \in T$.
- This assumption, by Theorem 2.1, is equivalent to (C2).
- The vector d is a regular direction if (R4) holds,
 - (R5*) the following second-order directional derivative exists for all $t \in T$:

$$g''(t, \hat{z}, d) := \lim_{\varepsilon \rightarrow 0+} 2 \frac{g(t, \hat{z} + \varepsilon d) - g(t, \hat{z}) - \varepsilon g'(t, \hat{z})d}{\varepsilon^2},$$

and the limit is uniform in t ; furthermore, H satisfies the same assumption as in (R5).

- (R6*) $\sigma(g(\cdot, \hat{z}), g'(\cdot, \hat{z})d|Q(\cdot))(t, \xi) > -\infty$ for $t \in T$, $\xi \in \mathbf{R}^r$, where the function $\sigma(x, d|Q(\cdot)) = \sigma(x, d|Q)$ is defined in (3.3).

It is easy to see that (R5*) yields that the map G defined in (5.1) satisfies the condition (R5). The regularity assumption made in (R6*) is also equivalent to (R6) by Theorem 3.5.

After these considerations, the main result of the section is the following theorem. Its proof employs the results derived in section 3.

THEOREM 5.1. *Let $\hat{z}$ be a regular solution of the above problem $(\mathcal{P}^*)$. Then, for all regular critical directions d , there exist Lagrange multipliers $\lambda \geq 0$, a signed vector-valued measure $\mu \in \mathcal{M}(T, \mathbf{R}^r)$, and $y^* \in Y^*$ such that at least one of them is different from zero and the following relations hold:*

$$(5.2) \quad \frac{d\mu}{d|\mu|}(t) \in N(g(t, \hat{z})|Q(t)) \quad \text{for } \mu\text{-a.e. } t \in T,$$

$$(5.3) \quad \lambda f^o(\hat{z}; z) + \int_T g'(t, \hat{z})z d\mu(t) + \langle y^*, H'(\hat{z})z \rangle \geq 0 \quad \text{for } z \in Z,$$

and

$$(5.4) \quad \begin{aligned} & \lambda f^{oo}(\hat{z}, d) + \int_T g''(t, \hat{z}, d) d\mu(t) + \langle y^*, H''(\hat{z}, d) \rangle \\ & \geq 2 \int_T \overline{\text{co}} \sigma(g(\cdot, \hat{z}), g'(\cdot, \hat{z})d|Q) \left(t, \frac{d\mu}{d|\mu|}(t) \right) d|\mu|(t). \end{aligned}$$

Proof. As we have already verified, the assumptions of Theorem 1.1 follow from that of Theorem 5.1 if we define G and $\mathbf{Q}$ by (5.1). Thus it suffices to show that the conclusions of Theorem 1.1 yield that of Theorem 5.1.

The existence of the Lagrange multiplier $x^* \in X^* = (C(T, \mathbf{R}^r))^*$ is equivalent to the existence of the measure $\mu \in \mathcal{M}(T, \mathbf{R}^r)$ by Riesz's representation theorem. Applying Theorems 2.1 and 3.10, (5.2), (5.3), and (5.4) are equivalent to (1.1), (1.2), and (1.3), respectively. $\square$

We consider now a special case of Theorem 5.1 when the compact metric space consists of a single element:

$$(\mathcal{P}^{**}) \quad \text{minimize } f(z) \quad \text{subject to} \quad g(z) \in Q, \quad H(z) = 0.$$

Here $Q \subset \mathbf{R}^r$ is a closed convex set with nonempty interior and $g : D \rightarrow \mathbf{R}^r$ is a continuous function on D . The assumptions (R2*), (C2*), (R5*), and (R6*) of problem $(\mathcal{P}^*)$ have to be replaced by

(R2**) The map $g(\cdot)$ is strictly Fréchet differentiable at $\hat{z}$;

(C2**) $g'(\hat{z})d \in T(g(\hat{z})|Q)$;

(R5**) the following second-order directional derivative exists

$$g''(\hat{z}, d) := \lim_{\varepsilon \rightarrow 0+} 2 \frac{g(\hat{z} + \varepsilon d) - g(\hat{z}) - \varepsilon g'(\hat{z})d}{\varepsilon^2};$$

(R6**) $\sigma(g(\hat{z}), g'(\hat{z})d|Q)(\xi) > -\infty$ for $\xi \in \mathbf{R}^r$.

COROLLARY 5.2. *Let $\hat{z}$ be a regular solution of the above problem $(\mathcal{P}^{**})$. Then, for all regular critical directions d , there exist Lagrange multipliers $\lambda \geq 0$, a vector $\mu \in \mathbf{R}^r$, and $y^* \in Y^*$ such that at least one of them is different from zero and the following relations hold:*

$$(5.5) \quad \mu \in N(g(\hat{z})|Q),$$

$$(5.6) \quad \lambda f''(\hat{z}; z) + \mu^T g'(\hat{z})z + \langle y^*, H'(\hat{z})z \rangle \geq 0 \quad \text{for } z \in Z,$$

and

$$(5.7) \quad \lambda f^{oo}(\hat{z}, d) + \mu^T g''(\hat{z}, d) + \langle y^*, H''(\hat{z}, d) \rangle \geq 2 \overline{\text{co}} \sigma(g(\hat{z}), g'(\hat{z})d|Q)(\mu).$$

Proof. Let $T = \{0\}$ and $\tilde{g} : T \times D \rightarrow \mathbf{R}^r$ be defined by $\tilde{g}(t, z) := g(z)$, and let $\tilde{Q}(\cdot) : T \rightarrow 2^{\mathbf{R}^r}$ be given by $\tilde{Q}(t) = Q$. Making these substitutions, problem $(\mathcal{P}^{**})$ reduces to $(\mathcal{P}^*)$, where g and $Q(\cdot)$ are replaced by $\tilde{g}$ and $\tilde{Q}(\cdot)$, respectively. The assumptions indicated by ** yield the analogous ones marked by *. Thus the conclusion of Theorem 5.1 is valid. Observe that, in this case, a vector-valued measure on T can be identified with a single vector in $\mathbf{R}^r$. Thus the conclusion of Theorem 5.1 reduces to that of Corollary 5.2. $\square$

We consider now another special case of Theorem 5.1, when $Q(t) =]-\infty, 0]$ for all $t \in T$. That is we investigate the following problem:

$$(\mathcal{P}^{***}) \quad \text{minimize } f(z) \quad \text{subject to} \quad g(t, z) \leq 0 \quad (\forall t \in T) \quad H(z) = 0.$$

Here $g : T \times D \rightarrow \mathbf{R}$ is a continuous function. The assumptions (C2*) and (R6*) of problem $(\mathcal{P}^*)$ have to be replaced by

(C2***) $g'(t, \hat{z})d(t) \leq 0$ if $g(t, \hat{z}) = 0$;

(R6***) $\sigma_{g(\cdot, \hat{z}), g'(\cdot, \hat{z})d(\cdot)}(t) > -\infty$ for $t \in T$.

COROLLARY 5.3. *Let $\hat{z}$ be a regular solution of the above problem $(\mathcal{P}^{***})$. Then, for all regular critical directions d , there exist Lagrange multipliers $\lambda \geq 0$, a nonnegative (scalar-valued) measure $\mu \in \mathcal{M}(T, \mathbf{R})$, and $y^* \in Y^*$ such that at least one of them is different from zero and the following relations hold:*

$$(5.8) \quad \text{supp } \mu \subset \{ t \in T \mid g(t, \hat{z}) = 0 \},$$

$$(5.9) \quad \lambda f^o(\hat{z}; z) + \int_T g'(t, \hat{z}) z d\mu(t) + \langle y^*, H'(\hat{z})z \rangle \geq 0 \quad \text{for } z \in Z,$$

and

$$(5.10) \quad \lambda f^{oo}(\hat{z}, d) + \int_T g''(t, \hat{z}, d) d\mu(t) + \langle y^*, H''(\hat{z}, d) \rangle \geq 2 \int_T \sigma_{g(\cdot, \hat{z}), g'(\cdot, \hat{z})d(\cdot)}(t) d\mu(t).$$

Proof. The equivalence of conditions (5.4) and (5.10) follows from Lemma 4.1 and Corollary 4.2. Condition (5.2) can be translated into the following form:

$$\frac{d\mu}{d|\mu|}(t) \begin{cases} \geq 0 & \text{if } g(t, \hat{z}) = 0, \\ = 0 & \text{if } g(t, \hat{z}) < 0. \end{cases}$$

Therefore, μ is nonnegative valued and (5.8) holds. $\square$

To conclude this section, we present a numerical example illustrating how the function σ can be explicitly calculated. The problem considered in the example can be handled with classical results as well. Our intention is to demonstrate how our results provide a new method for computing a second-order optimality criterion that does not require the computation of the set of the second-order tangent variations. Instead, we only need to calculate the function σ . For problems that cannot be handled without using the envelope-effect theory, we refer the readers to the papers of Kawasaki [6], [7], [8].

Example. Let Q be the closed unit disk in $\mathbf{R}^2$ and $f(z_1, z_2) := -z_2^2 - 2z_1$ for $(z_1, z_2) \in \mathbf{R}^2$. We consider the following optimization problem: minimize f on Q . As we shall see, the point $\hat{z} = (1, 0)$ satisfies the first- and second-order necessary conditions of Corollary 5.2. (Here we take the function $g = \text{id}$ and the vector space $Y = \{0\}$ and $H = 0$.)

It is easy to see that $\hat{z}$ is an admissible regular point for this problem. A vector $d = (d_1, d_2)$ is a critical direction if

$$f'(\hat{z})d \leq 0 \quad \text{and} \quad d \in T(\hat{z}|Q),$$

that is, if $d_1 = 0$. Therefore, $d = (0, 1)$ is a (typical) critical direction. We show that it is also regular. We have to check condition $(R6^{**})$. Observe that

$$\mathcal{C}(\hat{z}, d|Q) = \{ (\xi_1, \xi_2) \mid \xi_1 \geq 0, \xi_2 = 0 \}$$

as it is defined in (3.2) (we simply omit t, T from the definitions, because they do not play any role). On the other hand, if $\xi = (\xi_1, \xi_2)$, then $\delta^*(\xi|Q) = |\xi|$. Thus, for $\mathcal{S}(\hat{z}, d|Q)$, we have

$$\mathcal{S}(\hat{z}, d|Q) = \{ (\xi_1, \xi_2) \mid \xi_2 > 0 \}.$$

Clearly, $\mathcal{C}(\hat{z}, d|Q) \subset \mathcal{S}'(\hat{z}, d|Q)$. Therefore, if $\xi = (\xi_1, 0) \in \mathcal{C}(\hat{z}, d|Q)$, then

$$\begin{aligned}\sigma(\hat{z}, d|Q)(\xi) &= \liminf_{\substack{(\zeta_1, \zeta_2) \rightarrow (\xi_1, 0) \\ \zeta_2 > 0}} \frac{\zeta_2^2}{4[\zeta_1 - \sqrt{\zeta_1^2 + \zeta_2^2}]} \\ &= \liminf_{\substack{(\zeta_1, \zeta_2) \rightarrow (\xi_1, 0) \\ \zeta_2 > 0}} \frac{\zeta_2^2(\zeta_1 + \sqrt{\zeta_1^2 + \zeta_2^2})}{-4\zeta_2^2} = \frac{-\xi_1}{2} > -\infty,\end{aligned}$$

and $\sigma(\hat{z}, d|Q) = +\infty$ otherwise. We have checked that $(R6^{**})$ holds, too. Furthermore, we also see that $\sigma(\hat{z}, d|Q) = \overline{\text{co}} \sigma(\hat{z}, d|Q)$ in this case.

To complete this argument, we have to show that there exist $\lambda \geq 0$ and $\mu = (\mu_1, \mu_2)$ such that (5.5), (5.6), and (5.7) hold. It follows from (5.6) that $\mu_2 = 0$ and $\mu_1 \geq 0$. The inequality in (5.6) is satisfied if and only if

$$\lambda \partial_1 f(\hat{z}) + \mu_1 = -2\lambda + \mu_1 = 0 \quad \text{and} \quad \lambda \partial_2 f(\hat{z}) = 0.$$

If λ were 0, then, by the first equation, $\mu_1 = 0$, which contradicts $\lambda + |\mu| \neq 0$. Thus we may assume that $\lambda = 1$. Then $\mu_1 = 2$. We have to check whether (5.5) is also satisfied. For the right-hand side of (5.7) we have $2\sigma(\hat{z}, d|Q)(2, 0) = -2$. On the other hand,

$$f^{oo}(\hat{z})(d) = d^T f''(\hat{z})d = \partial_2^2 f(\hat{z}) = -2,$$

which shows that (5.7) is satisfied by equality.

It is also easy to see that f attains minimum on Q at $\hat{z} = (0, 1)$, because, for $|z| \leq 1$,

$$f(z) = -z_2^2 - 2z_1 \geq -1 + z_1^2 - 2z_1 = (z_1 - 1)^2 - 2 \geq -2 = f(1, 0).$$

REFERENCES

- [1] F. H. CLARKE, *Optimization and Nonsmooth Analysis*, Canadian Math. Soc. Ser. Monogr. Adv. Texts, John Wiley, New York, 1983.
- [2] R. COMINETTI, *Metric regularity, tangent sets and second-order optimality conditions*, Appl. Math. Optim., 21 (1990), pp. 265–287.
- [3] A. YA. DUBOVITSKII AND A. A. MILYUTIN, *Extremum problems with constraints*, Dokl. Akad. Nauk SSSR, 149 (1963), pp. 759–762; Soviet Math. Dokl., 4 (1963), pp. 452–455.
- [4] I. V. GIRSANOV, *Lectures on Mathematical Theory of Extremum Problems*, Lecture Notes in Econom. and Math. Systems 67, Springer-Verlag, New York, 1972.
- [5] A. D. IOFFE, *On some recent developments in the theory of second-order optimality conditions*, in Optimization, S. Dolecki, ed., Fifth French-German Conference Castel-Novel, 1988. Lecture Notes in Math. 1405, Springer-Verlag, New York, 1989, pp. 55–68.
- [6] H. KAWASAKI, *An envelope-like effect of infinitely many inequality constraints on second-order necessary conditions for minimization problems*, Math. Programming, 41 (1988), pp. 73–96.
- [7] H. KAWASAKI, *The upper and lower second order directional derivatives of a sup-type function*, Math. Programming, 41 (1988), pp. 327–339.
- [8] H. KAWASAKI, *Second order necessary optimality for minimizing a sup-type function*, Math. Programming, 49 (1991), pp. 213–229.
- [9] Y. MARUYAMA, *Second-order necessary conditions for nonlinear optimization problems in Banach spaces by the use of Neustadt derivative*, Math. Japonica, 40 (1994), pp. 509–522.
- [10] P. D. LOEWEN AND R. T. ROCKAFELLAR, *The adjoint arc in nonsmooth optimization*, Trans. Amer. Math. Soc., 325 (1991), pp. 39–72.
- [11] ZS. PÁLES AND V. M. ZEIDAN, *Nonsmooth optimum problems with constraints*, SIAM J. Control Optim., 32 (1994), pp. 1476–1502.
- [12] ZS. PÁLES AND V. M. ZEIDAN, *Characterization of closed C-convex sets in $C(T, \mathbf{R}^n)$* , Acta Sci. Math. Szeged, submitted.

- [13] J.-P. PENOT, *Optimality conditions in mathematical programming and composite optimization*, Mathematical Programming 67 (1994), pp. 225–245.
- [14] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
- [15] R. T. ROCKAFELLAR, *Integrals which are convex functionals*, Pacific J. Math., 24 (1968), pp. 525–539.
- [16] R. T. ROCKAFELLAR, *Integrals which are convex functionals*, II, Pacific J. Math., 39 (1971), pp. 439–469.

APPLYING A NEWTON METHOD TO STRICTLY CONVEX SEPARABLE NETWORK QUADRATIC PROGRAMS*

J. SUN[†] AND H. KUO[‡]

Abstract. By introducing quadratic penalty terms, a strictly convex separable network quadratic program can be reduced to an unconstrained optimization problem whose objective is a continuously differentiable piecewise quadratic function. A recently developed nonsmooth version of Newton's method is applied to the reduced problem. The generalized Newton direction is computed by an iterative procedure which exploits the special network data structures that originated from the network simplex method. New features of the algorithm include the use of min-max bases and a dynamic strategy in computation of the Newton directions. Some preliminary computational results are presented. The results suggest the use of "warm start" instead of "cold start."

Key words. network quadratic programming, Newton's method, nonsmooth optimization, penalty method, piecewise quadratic programming

AMS subject classifications. 65K05, 90B10, 90C20, 90C30, 90C35

PII. S1052623494265359

1. Introduction.

1.1. Motivation of this research. Convex separable network quadratic programming finds its applications in various engineering, economic, and business fields such as electrical circuits [13, 20], stick percolation [12], and portfolio management [24, 26]. Quite often, it also appears as a subproblem in solving more general nonlinear network programming problems. It is not surprising, therefore, that many algorithms have been developed to solve it. Among them are the Frank–Wolfe algorithm [14], the convex simplex method [8], the active set method [2, 19], the relaxation method [1], the fortified algorithm [31], the Newton-type algorithms [10, 22], and the reduced gradient algorithm [11], to name just a few.

This paper is not aimed at developing a new method for network quadratic programming and comparing its performance with the existing algorithms. Rather, we have the following considerations in mind. First, the algorithm developed here should become a subroutine that can be used in the future to solve *nonsmooth* network programming problems, so it should be readily modified to allow separable *piecewise* quadratic costs; second, we want to test the idea of nonsmooth penalties in network programming, which has been shown promising in the research of Conn [9] and Gamble, Conn, and Pulleyblank [16] for the linear cases; finally, we want to investigate the usefulness of the recently developed nonsmooth Newton method [28] using network quadratic programming as a work site.

From our point of view the major advantage of Newton's method is its convergence rate, so one would benefit from it especially at the last few iterations of computation. The major disadvantage of it is that the computation of an exact Newton direction is very costly; in the case of network programming this computation may be prohibitive

* Received by the editors March 21, 1994; accepted for publication (in revised form) July 24, 1997; published electronically June 3, 1998. This research was partially supported by grant RP-930033 of the National University of Singapore.

<http://www.siam.org/journals/siopt/8-3/26535.html>

[†] Department of Decision Sciences, National University of Singapore, 119260 Singapore (fbasunj@nus.edu.sg).

[‡] 36 Yin-Lian Alley 2, Taichun, Taiwan.

because of the size of the problems and the hardship of incorporating network structures. However, it might be possible to compute an approximate Newton direction iteratively by exploiting the efficient data structures of networks. It seems to be worthwhile to pursue a study of this aspect.

1.2. Problem formulation. A directed network consists of m nodes and n arcs. Each arc j originates at some node i and terminates at another node i' . In a network flow model a certain commodity travels along the arcs in order to satisfy the supply or demand requirements at the nodes. It is convenient to think of demand as negative supply in this case. Let s_i , $i = 1, \dots, m$, be the (positive or negative) supply at node i , and let the total amount of the commodity that travels along a particular arc j be x_j , $j = 1, \dots, n$. We call x_j the *flow* on arc j and call

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad \text{and} \quad s = \begin{pmatrix} s_1 \\ \vdots \\ s_m \end{pmatrix}$$

the *flow vector* and the *supply vector*, respectively. We define the node-arc incidence matrix by

$$E = \{e(i, j)\}_{i=1}^m \quad \begin{matrix} n \\ j=1 \end{matrix},$$

where

$$e(i, j) = \begin{cases} +1 & \text{if } i \text{ is the initial node of the arc } j, \\ -1 & \text{if } i \text{ is the terminal node of the arc } j, \\ 0 & \text{otherwise.} \end{cases}$$

Note that by the definitions of E and x the i th entry of the vector Ex represents the total supply of x at node i . A convex separable network quadratic program seeks the best flow vector x to minimize the total quadratic costs incurred on the arcs, subject to the outside supply requirements at the nodes as well as bound constraints for the flow vector. The mathematical form of the problem is as follows:

$$(\text{NQP}) \quad \begin{cases} \text{minimize} & p(x) \equiv \sum_{j=1}^n (\frac{1}{2} p_j x_j^2 + c_j x_j) \equiv \frac{1}{2} x^T P x + c^T x \\ \text{subject to} & Ex = s, \\ & l \leq x \leq u, \quad x \in R^n, \end{cases}$$

where the superscript " T " denotes the transpose, s is the fixed supply vector, the vector inequalities are understood coordinatewise, $c, l, u \in R^n$ are fixed vectors, and P is an $n \times n$ diagonal matrix with nonnegative diagonal elements. If P is positive definite, then the objective function is strictly convex. In this paper we concentrate on this case for simplicity of discussion. As a matter of fact, quite a few principles used in our analysis can be extended to general convex cases as well.

1.3. The data structures of networks. Rich combinatorial structures exist in the network flow models in connection with the simplex method. We briefly describe the structures related to our task.

Without loss of generality, we assume that the network in problem (NQP) is connected. It is not hard to show that in this case the system $Ex = s$ contains exactly $m - 1$ independent equations. Arbitrarily deleting an equation from the system, we can equivalently write $Ex = s$ as

$$(1) \quad Bx_B + Nx_N = b \quad \text{or} \quad x_B = Rx_N + \bar{b},$$

where B is an $(m-1) \times (m-1)$ (invertible) basis matrix, N is an $(m-1) \times (n-m+1)$ nonbasis matrix, b is obtained by deleting the corresponding entry from s , $R = -B^{-1}N$, $\bar{b} = B^{-1}b$, and

$$x = \begin{pmatrix} x_B \\ x_N \end{pmatrix}$$

is a basis-nonbasis partition of the variables. The arcs corresponding to x_B form a spanning tree of the network rooted at the node corresponding to the deleted equation. In addition, for any basis B in (1) there exists an ordering of the rows and columns such that B is upper triangular. It is therefore very easy to compute $B^{-1}y$ and $B^{-T}y$ for any given vector y . These computations can be performed efficiently by using the following three arrays.

The predecessor. The first node on the path in the spanning tree from node i to the root of the tree.

Notice that when the tree is drawn as hanging down from the root, the predecessor of each node is its unique neighbor on the next higher level. In the case of a spanning tree of a network, every node except the root node in the network has a predecessor. The main virtue of the predecessor array is that it can store compactly all the paths between the root and other nodes in the tree.

The depth. The number of nodes on the path between a node i in the tree and the root node, including i and the root.

Using the depth and the predecessor, we can quickly identify the cycle formed by the addition of an arc to the spanning tree.

The successor. A chainwise structure (or thread) in the nodes according to a certain ordering relation.

The successor actually stores the triangular structure of B . It can be used to compute Ry and $R^T y$ in combination with the depth and the predecessor. For more details of the usage of the three arrays in the context of the network simplex method see [6] or [21]. For a discussion of the structure of matrix R see [31].

1.4. A nonsmooth version of Newton's method. We now describe some basic results on nonsmooth analysis and a nonsmooth version of Newton's methods. A function $F : X \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ is called *locally Lipschitzian* on X if for any neighborhood $N \subset X$ there exists a constant L such that for any $x, y \in N$, one has

$$\|F(x) - F(y)\| \leq L\|x - y\|.$$

A locally Lipschitzian function on X is differentiable almost everywhere in X (the Rademacher theorem). Denote the set of points at which F is differentiable by D_F . We write $JF(x)$ for the usual $n \times n$ Jacobian matrix of partial derivatives whenever x is a point at which the necessary partial derivatives exist. Let $\partial F(x)$ be the generalized Jacobian defined by Clarke (in section 2.6 of Clarke [7]). That is,

$$(2) \quad \partial F(x) = \text{co} \left\{ \lim_{\substack{x' \rightarrow x \\ x' \in D_F}} JF(x') \right\},$$

where “co” denotes the convex hull in the usual sense of convex analysis (cf. [30]).

We say that F is *semismooth* at x if F is locally Lipschitzian at x and

$$(3) \quad \lim_{\substack{V \in \partial F(x+th') \\ h' \rightarrow h, t \downarrow 0}} \{Vh'\}$$

exists for any $h \in R^n$. Semismoothness was originally introduced by Mifflin [25] for functionals. Convex functions, smooth functions, and subsmooth functions are examples of semismooth functions. Composition with affine transformations, scalar products, and sums of semismooth functions are still semismooth functions.

Now think of $F(x)$ as the gradient function of a continuously differentiable convex function $f(x)$. In this case a member in the set of $\partial F(x)$ is naturally called a *generalized Hessian* of f at x . The problem of minimizing $f(x)$ over R^n is then theoretically equivalent to finding a root of $F(x)$. Suppose that F is semismooth. We wish to find the solution of the equation

$$(4) \quad F(x) = 0.$$

Let us consider the following nonsmooth version of Newton's method

$$(5) \quad x^{k+1} = x^k - V_k^{-1} F(x^k),$$

where $V_k \in \partial F(x^k)$.

The groundwork on the convergence of the method (5) has been laid in Qi and Sun [28]. The following theorem summarizes the result.

THEOREM 1.1. *Suppose that F is semismooth at x^* , all $V \in \partial F(x^*)$ are nonsingular, and certain other regularity conditions hold. Then the iteration method (5) is well defined and globally convergent to the unique solution x^* of (4) at a q -superlinear rate.*

An adjustment is needed to ensure a monotonically decreasing sequence $\{f(x^k)\}$ when the method (5) is applied to the problem of minimizing $f(x)$. Instead of using (5), we use

$$(6) \quad x^{k+1} = x^k + \beta_k \Delta x^k \text{ with } \Delta x^k = -V_k^{-1} F(x^k),$$

where β_k is a suitable stepsize obtained via a line search procedure. The most commonly used line search strategy is the so-called Goldstein–Armijo line search (cf. [17]). However, if the objective function is piecewise quadratic in the sense that $f(x)$ is a convex function whose domain is a union of finitely many convex polyhedra, on each of which $f(x)$ is given by a quadratic formula of x , as we will see in the next section, it is fairly easy to conduct the exact line search, so no Goldstein–Armijo line search is necessary. In addition, the computation of V_k in (6) becomes extremely simple if $f(x)$ is piecewise quadratic. In this case by definition R^n is the union of a finite number of polyhedra on each of which $f(x)$ is given by a quadratic formula. Let $f(x) = \frac{1}{2}x^T Qx + d^T x + \gamma$ in a polyhedron containing x^k . (Note that the definition of piecewise quadratic functions does not exclude the possibility that the value of $f(x^k)$ could be given by different formulae if x^k simultaneously belongs to several polyhedra in the domain.) Then $F(x) \equiv \nabla f(x)$ and $Q \in \partial F(x^k)$. Therefore, we may just take Q as V_k . For example, consider the one-dimensional piecewise quadratic function

$$f(x) = \begin{cases} x^2 & \text{if } x < 0, \\ 2x^2 & \text{if } x \geq 0. \end{cases}$$

The Hessian of $f(x)$ is 2 for $x < 0$ and 4 for $x > 0$, and so is the generalized Hessian of $f(x)$. At $x = 0$, $f(0)$ can be computed by either formula. The set of generalized Hessians $\partial F(0)$ is the closed interval $[2, 4]$ according to (2). Therefore V_k could be any value in $[2, 4]$. In particular the Hessian of x^2 or $2x^2$ at 0, i.e., 2 or 4, can be taken as V_k .

A further adjustment to the nonsmooth Newton method is made in the presence of equation constraints. Suppose that we want to

$$(7) \quad \text{minimize } f(x) \quad \text{subject to } Ex = s.$$

Then the nonsmooth Newton direction Δx^k in (6) is obtained by solving the quadratic program

$$(8) \quad \begin{cases} \text{minimize } \nabla f(x^k)^T \Delta x + \frac{1}{2} \Delta x^T V_k \Delta x \\ \text{subject to } E \Delta x = 0, \end{cases}$$

where V_k is any generalized Hessian of f at x^k .

It can be shown that $\nabla f(x)$ is semismooth if $f(x)$ is continuously differentiable and piecewise quadratic. Thus a similar result to Theorem 1.1 might be expected for problem (7). However, if $f(x)$ is in addition strictly convex, then a stronger result holds. Global and finite convergence of the algorithm using (6) and (8) for strictly convex, continuously differentiable, and piecewise quadratic problems with linear equality constraints has been proved in [33]. The result can be stated as follows.

THEOREM 1.2. *Let $X = \{x | Ex = s\}$ be nonempty, let $f(x)$ be strictly convex, continuously differentiable, and piecewise quadratic, and let Δx^k be an optimal solution to (8). Suppose that the line search is performed exactly, i.e.,*

$$x^{k+1} = x^k + \beta_k \Delta x^k \text{ such that } f(x^{k+1}) = \min_{\beta \geq 0} f(x^k + \beta \Delta x^k).$$

Then, starting from any $x^1 \in X$, the algorithm generates a finite sequence $\{x^k\}$, $k = 1, 2, \dots, t$, such that $x^t = x^*$, where x^* is the unique minimum of problem (7).

2. Development of the algorithm. In this section we describe the details of the method in Theorem 1.2 applied to problem (NQP). We first remove the bound constraints $l \leq x \leq u$ of (NQP) by incorporating them into the objective function as quadratic penalty terms. By doing this, problem (NQP) is reduced to an equality-constrained network piecewise quadratic program (NPQP), which is in the form of (7). The latter problem is then solved by the method stated in Theorem 1.2. The solution Δx^k to (8) is computed by an iterative procedure which exploits the special network data structures introduced in section 1.3.

The advantage of the quadratic penalty terms is that the new objective function will be strictly convex and continuously differentiable (which may not be satisfied by linear penalty terms, for instance). Hence Theorem 1.2 applies. It should also be noted that the nonsmooth Newton algorithm does not depend on any active constraint strategy. Thus this method can avoid excess computing time in degenerate cases, which may occur frequently in network problems.

In the following sections we will next derive some theoretical results for the penalty method in section 2.1. In section 2.2, we will discuss an iterative procedure for the computation of the Newton direction. Section 2.3 will include a min-max spanning tree strategy which may speed up the convergence of the procedure in section 2.2. The algorithm will be summarized together with a description of the line search process in section 2.4. Finally, we will address the specific modifications that have to be made, if the original objective function is piecewise quadratic, in section 2.5.

2.1. The penalty terms. Consider the problem

$$(NPQP) \quad \begin{cases} \text{minimize} & f(x) \equiv p(x) + \frac{\alpha}{2} \sum_{j=1}^n [\min^2\{x_j - l_j, 0\} + \min^2\{u_j - x_j, 0\}] \\ \text{subject to} & Ex = s. \end{cases}$$

This problem is obtained by dropping the constraints $l \leq x \leq u$ and adding penalty terms $\frac{1}{2}\alpha \sum_{j=1}^n [\min^2\{x_j - l_j, 0\} + \min^2\{u_j - x_j, 0\}]$ in (NQP), where $\alpha > 0$ is the penalty parameter.

A solution x is δ -feasible to (NQP) if $Ex = s$ and $\min\{x_j - l_j, u_j - x_j\} \geq -\delta$. A δ -feasible solution is ε -optimal to (NQP) if $p(x)$ is within ε of the optimal value of (NQP). In practice we start from an experimental value of α , say, $100\max\{p_1, \dots, p_n\}$. If the solution is not δ -feasible and ε -optimal to (NQP), then in the next iteration we replace α with 10α and repeat the algorithm until the solution is δ -feasible and ε -optimal to (NQP). This scheme is suggested in [16]. Now we prove a theorem which ensures that the process of enlarging α will end up with a δ -feasible and ε -optimal solution in finitely many iterations. For simplicity of notation we discuss a simple case where the bound constraints are actually $x \geq 0$. The general case can be proved similarly. The result is slightly more general than we need since we do not assume strict convexity of $p(x)$ in the proof.

THEOREM 2.1. *Suppose that problem (NQP) has a solution. For any $\delta > 0$ and $\varepsilon > 0$ there exists an $\bar{\alpha} > 0$ such that for $\alpha > \bar{\alpha}$ there is an optimal solution of (NPQP) which is also a δ -feasible and ε -optimal solution of (NQP).*

Proof. Recall that the objective functions of (NQP) and (NPQP) are $p(x)$ and $f(x)$, respectively. Define the optimal value function

$$\rho(u) = \min\{p(x) | Ex = s, x \geq -u\}.$$

Note that $\rho(0)$ is the optimal value of (NQP). According to Sun [32], $\rho(u)$ is a proper, convex, and piecewise quadratic function of u .

Let $e = (1, \dots, 1)^T$. Given $\delta > 0$ and $\varepsilon > 0$, we set $\bar{\alpha}$ such that for any $\alpha > \bar{\alpha}$,

$$(9) \quad \rho(u) + \frac{\alpha}{2} \sum_{j=1}^n u_j^2 > \rho(0) - \varepsilon,$$

and for any $\alpha > \bar{\alpha}$ and $u \geq \delta e$,

$$(10) \quad \rho(u) + \frac{\alpha}{2} \sum_{j=1}^n u_j^2 \geq \rho(0).$$

Since $\rho(u)$ is a proper convex piecewise quadratic function, such an $\bar{\alpha}$ exists. Note that

$$\begin{aligned} & \min_x \{f(x) | Ex = s\} \\ &= \min_{x, u} \{f(x) | Ex = s, x \geq -u\} \\ &= \min_{x, u} \{f(x) | Ex = s, x \geq -u, u \geq 0\} \\ &= \min_{x, u} \left\{ p(x) + \frac{\alpha}{2} \sum_{j=1}^n (\min\{x_j, 0\})^2 | Ex = s, x \geq -u, u \geq 0 \right\} \\ &= \min_{x, u} \left\{ p(x) + \frac{\alpha}{2} \sum_{j=1}^n (\min\{-u_j, 0\})^2 | Ex = s, x \geq -u, u \geq 0 \right\} \\ &= \min_{u \geq 0} \left\{ \frac{\alpha}{2} \sum_{j=1}^n (\min\{-u_j, 0\})^2 + \rho(u) \right\} \\ (11) \quad &= \min_{u \geq 0} \left\{ \frac{\alpha}{2} \sum_{j=1}^n u_j^2 + \rho(u) \right\}. \end{aligned}$$

Thus for any $\alpha > \bar{\alpha}$ we have

$$\begin{aligned}\rho(0) &\geq \min\{f(x)|Ex = s\} \\ &= \min_{u \geq 0} \left\{ \rho(u) + \frac{\alpha}{2} \sum_{j=1}^n u_j^2 \right\} && \text{(by using (11))} \\ &\geq \rho(0) - \varepsilon. && \text{(by using (9))}\end{aligned}$$

Therefore, any optimal solution of (NPQP) is ε -optimal to (NQP).

Now we show that

$$\min\{f(x)|Ex = b\} = \min\{f(x)|Ex = s, x \geq -\delta e\}.$$

Similar to the derivation of (11), we can show that

$$\begin{aligned}&\min\{f(x)|Ex = s, x \geq -\delta e\} \\ &= \min_{\delta e \geq u \geq 0} \left\{ \rho(u) + \frac{\alpha}{2} \sum_{j=1}^n u_j^2 \right\} \\ &= \min_{u \geq 0} \left\{ \rho(u) + \frac{\alpha}{2} \sum_{j=1}^n u_j^2 \right\} && \text{(by using (10))} \\ &= \min\{f(x)|Ex = s\}. && \text{(by using (11))}\end{aligned}$$

Therefore, if α is large enough, there is at least one optimal solution of (NPQP), which is also a δ -feasible and ε -optimal solution of (NQP). $\square$

It can be seen that if $f(x)$ is in particular strictly convex, then the unique solution of (NPQP) will be a δ -feasible and ε -optimal solution of (NQP).

2.2. The Newton method for the equality constrained problem. Now we consider how to solve problem (8). As mentioned in section 1.4, for the piecewise quadratic function $f(x)$ the Hessian of any formula that represents f at x^k can be taken as V_k in (8). Let such a Hessian be Q . Then the objective function in (8) is

$$(12) \quad \frac{1}{2} \Delta x^T Q \Delta x + \nabla f(x^k)^T \Delta x,$$

where Q and $\nabla f(x^k)$ can be determined by $\alpha, p_j, c_j, l_j, u_j$, and x^k . In order to find the search direction Δx^k in (8) we write the equation $E\Delta x = 0$ in its equivalent form in view of (1):

$$(13) \quad B\Delta x_B + N\Delta x_N = 0 \quad \text{or} \quad \Delta x_B = R\Delta x_N,$$

where R is an $(m-1) \times (n-m+1)$ matrix consisting of only 1, 0, and -1 (cf. section 4H of [31]). The basic and nonbasic directions Δx_B and Δx_N are in the same sense as in section 1.3. Correspondingly let

$$Q = \begin{pmatrix} Q_B & 0 \\ 0 & Q_N \end{pmatrix} \quad \text{and} \quad g = \nabla f(x^k) = \begin{pmatrix} g_B \\ g_N \end{pmatrix},$$

and substitute (13) into (12). The nonbasic component Δx_N of Δx^k can be found by solving the Newton equation

$$(14) \quad (Q_N + R^T Q_B R) \Delta x_N = -(R^T g_B + g_N) \equiv -r_N.$$

It is difficult to solve (14) directly because the matrix $Q_N + R^T Q_B R$ could be dense and large. However, the form of (14) naturally suggests an iterative process:

$$(15) \quad Q_N \Delta x_N^{t+1} = -R^T Q_B R \Delta x_N^t - r_N.$$

Given Δx_N^t , the right-hand side of equation (15) can be efficiently computed by taking advantage of the network structures while the computation of Δx_N^{t+1} makes no additional burden because of the diagonality of Q_N . In order to certify the convergence of procedure (15) we introduce a positive parameter λ and use the following forward-backward splitting algorithm:

$$(16) \quad (\lambda I + Q_N) \Delta x_N^{t+1} = (\lambda I - R^T Q_B R) \Delta x_N^t - r_N,$$

where $t = 1, 2, \dots$ and I is the unit matrix.

Some general theories on the convergence of forward-backward splitting algorithms have been established (cf. [4, 5, 15]). We now offer a special proof for (16) which may lead to a convergence rate related to the network structure and thus suggests some useful heuristics in the design of the algorithm.

THEOREM 2.2. *Let $M = (\lambda I + Q_N)^{-1}(\lambda I - R^T Q_B R)$, $\lambda > 0$. Suppose that Q_B , Q_N are positive semidefinite and suppose that $0 \leq \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_{n-m+1}$ are the eigenvalues of the matrix $R^T Q_B R$. Then a sufficient condition for the convergence of procedure (16) is*

$$(\lambda + q_{N_{\min}})^{-1} \max\{|\lambda - \lambda_1|, |\lambda - \lambda_{n-m+1}|\} < 1,$$

where $q_{N_{\min}}$ is the smallest diagonal element of Q_N .

Proof. Let G be an orthogonal matrix such that

$$G^{-1}(R^T Q_B R)G = \begin{pmatrix} \lambda_1 & & \circ \\ & \ddots & \\ \circ & & \lambda_{n-m+1} \end{pmatrix}.$$

Then we have

$$\begin{aligned} \|\lambda I - R^T Q_B R\|_2 &= \|G^{-1}(\lambda I - R^T Q_B R)G\|_2 \\ &= \left\| \begin{pmatrix} \lambda - \lambda_1 & & \circ \\ & \ddots & \\ \circ & & \lambda - \lambda_{n-m+1} \end{pmatrix} \right\|_2 \\ &\leq \max\{|\lambda - \lambda_1|, |\lambda - \lambda_{n-m+1}|\}. \end{aligned}$$

On the other hand, there holds

$$\begin{aligned} \|(\lambda I + Q_N)^{-1}\|_2 &= \left\| \begin{pmatrix} \lambda + q_{N_1} & & \circ \\ & \ddots & \\ \circ & & \lambda + q_{N_{n-m+1}} \end{pmatrix}^{-1} \right\|_2 \\ &\leq (\lambda + q_{N_{\min}})^{-1}. \end{aligned}$$

Therefore, the spectral radius of M satisfies

$$\begin{aligned} \rho(M) &\leq \|M\|_2 \\ &\leq \|(\lambda I + Q_N)^{-1}\|_2 \|\lambda I - R^T Q_B R\|_2 \\ &\leq (\lambda + q_{N_{\min}})^{-1} \max\{|\lambda - \lambda_1|, |\lambda - \lambda_{n-m+1}|\} \\ &< 1. \end{aligned}$$

Hence the procedure (16) converges to the solution of system (14) (cf. Theorem 10.1-1 in [18]). $\square$

THEOREM 2.3. *Using the same notation as in Theorem 2.2, if the matrix E is an incidence matrix of some network, then one has*

$$\lambda_1 \leq \text{trace}(Q_B) \quad \text{and} \quad \lambda_{n-m+1} \leq (n-m+1)\text{trace}(Q_B),$$

where $\text{trace}(Q_B)$ is the trace of Q_B .

Proof. Consider the trace of $R^T Q_B R$. Since all entries of R are 1, -1 , or 0, every column of $Q_B R$ contains only elements like $\pm q_B$ and 0, where q_B denotes an arbitrary diagonal element of Q_B . Thus each diagonal entry of $R^T Q_B R$ cannot exceed $\text{trace}(Q_B)$. Then we obtain

$$\lambda_1 \leq \frac{\lambda_1 + \cdots + \lambda_{n-m+1}}{n-m+1} = \frac{\text{trace}(R^T Q_B R)}{n-m+1} \leq \text{trace}(Q_B)$$

and

$$\lambda_{n-m+1} \leq \lambda_1 + \cdots + \lambda_{n-m+1} = \text{trace}(R^T Q_B R) \leq (n-m+1)\text{trace}(Q_B). \quad \square$$

COROLLARY 2.4. *Suppose that $q_{N_{\min}} > 0$ or $\lambda_1 > 0$. Then for all $\lambda > (n-m+2)\text{trace}(Q_B)/2$ the procedure (16) converges to a solution of system (14).*

Proof. If $\lambda > (n-m+2)\text{trace}(Q_B)/2$, then we have

$$\lambda > \frac{1}{2}(\lambda_1 + \lambda_{n-m+1})$$

and

$$\begin{aligned} & (\lambda + q_{N_{\min}})^{-1} \max\{|\lambda - \lambda_1|, |\lambda - \lambda_{n-m+1}|\} \\ &= (\lambda + q_{N_{\min}})^{-1}(\lambda - \lambda_1) \\ &< 1. \end{aligned}$$

Thus the procedure (16) converges to a solution of system (14) according to Theorem 2.2. $\square$

Corollary 2.4 offers a safeguard value for the convergence of procedure (16), which is easy to compute but may not be the best in practice.

2.3. The use of min-max bases. In order to accelerate convergence, we propose a heuristic method to update λ in the iterative process. At the t th iteration, λ is computed as follows. If $t = 1$ then $\lambda = (n-m+2)\text{trace}(Q_B)/2$; if $t > 1$, then λ is the smaller of $(n-m+2)\text{trace}(Q_B)/2$ and

$$(17) \quad \frac{(\Delta x_N^{t-1})^T R^T Q_B R \Delta x_N^{t-1}}{\|\Delta x_N^{t-1}\|^2} = \left(\frac{\|\sqrt{Q_B} R \Delta x_N^{t-1}\|}{\|\Delta x_N^{t-1}\|} \right)^2,$$

where $\sqrt{Q_B}$ is the diagonal matrix whose diagonal elements are the square roots of the corresponding diagonal elements in Q_B .

This heuristic choice of λ is based on Theorem 2.2. We try to choose λ between λ_1 and λ_{n-m+1} so that $\max\{|\lambda - \lambda_1|, |\lambda - \lambda_{n-m+1}|\}$ is small. The Rayleigh quotient in (17) is definitely in the interval $[\lambda_1, \lambda_{n-m+1}]$, and its value can be obtained as a by-product of procedure (16). We also try to make $\lambda + q_{N_{\min}}$ large, which is equivalent

to minimizing the largest q_B among all possible spanning trees. Roughly speaking, if the elements in the diagonal of matrix Q_B are relatively small and the elements in the diagonal of Q_N are relatively large, it will accelerate the convergence of procedure (16). In practical tests we have also found that such min-max spanning trees perform better than the others.

DEFINITION 2.5. *A basis B is min-max if there is no alternative basis B' such that the greatest element in $Q_{B'}$ is less than the greatest element in Q_B .*

There are fast algorithms to generate the min-max spanning tree such as those introduced in [3]. We omit the details of the algorithms here.

In summary, the Newton directions of problem (NPQP) can be obtained by the following procedure.

PROCEDURE 2.6.

Step 1. *Given the current iterate x , create the min-max basis B in terms of Q .*

Step 2. *Compute r_N by formula (14).*

Step 3. *Arbitrarily choose $\Delta x_N^1 \neq 0$ and let $t = 1$.*

Step 4. *Compute $\sqrt{Q_B} R \Delta x_N^t$ and*

$$\lambda = \min \left\{ (n - m + 2) \text{trace}(Q_B) / 2, \frac{(\Delta x_N^{t-1})^T R^T Q_B R \Delta x_N^{t-1}}{(\Delta x_N^{t-1})^T (\Delta x_N^{t-1})} \right\}.$$

The initial λ can be $(n - m + 2) \text{trace}(Q_B) / 2$ as indicated by Corollary 2.4.

Step 5. *Compute $R^T Q_B R \Delta x_N^{t-1}$ and*

$$\Delta x_N^t = (\lambda I + Q_N)^{-1} [\lambda \Delta x_N^{t-1} - R^T Q_B R \Delta x_N^{t-1} - r_N].$$

Step 6. *Check the terminating criterion; if it is not satisfied, update t and return to Step 4; else go to Step 7.*

Step 7. *Let $\Delta x_N = \Delta x_N^t$, compute $\Delta x_B = R \Delta x_N$, and terminate.*

In this subroutine the total number of operations in each iteration is $O(n)$. The core step of the above procedure is the computation of $R \Delta x_N$ and $R^T y$ for a certain given vector y . Both reduce to the computation of $B^{-1}z$ or $B^{-T}z$ for a vector z , which can be done quickly by using the data structures in section 1.3. We omit the concrete steps since they are completely the same as in the network simplex method.

2.4. The algorithm and line searches. After the feasible search direction Δx is obtained, we start a line search to find the minimum of $f(x)$ along Δx . Since Δx is a Newton direction, if x is not optimal to (NPQP), we must have

$$\arg \min_{\beta \geq 0} f(x + \beta \Delta x) \equiv \beta^* > 0.$$

The fact that $\beta^* > 0$ is due to the descending property of the Newton direction and, together with the convexity of the function, it ensures that the global minimum of $f(x + \beta \Delta x)$ is in the direction of Δx . In addition, since $f(x + \beta \Delta x)$ is a one-dimensional piecewise quadratic function, the line search is pretty simple.

The way to compute β^* is as follows. Notice that the function $f(x + \beta \Delta x)$ can be expressed by

$$f(x + \beta \Delta x) = \frac{1}{2} \beta^2 (\Delta x)^T Q (\Delta x) + \beta g^T \Delta x + \text{constant}$$

in the interval $[0, \bar{\beta}]$, where $\bar{\beta}$ is the minimal steplength that makes one of the components of $x + \beta \Delta x$ hit its bound at which the piecewise quadratic objective function

“changes piece.” If the “nominal” minimum $-g^T \Delta x / [(\Delta x)^T Q(\Delta x)]$ is not larger than $\bar{\beta}$, then the nominal minimum is the real minimum of $f(x + \beta \Delta x)$. Otherwise, we change Q and g at the boundary point $\bar{\beta}$, compute the next $\bar{\beta}$ and nominal minimum, and repeat the procedure. This procedure ends up with an optimum in a finite number of steps since $f(x) \rightarrow \infty$ as $\|x\| \rightarrow \infty$.

Now we summarize the algorithm for (NQP).

THE NONSMOOTH NEWTON METHOD FOR (NQP)

Step 0. (Initialization) Arbitrarily choose x^0 such that $Ex^0 = s$, select a starting penalty parameter α , and set the user-assigned optimality tolerances $\delta > 0$ and $\epsilon > 0$. Set $k = 0$.

Step 1. (The Newton Iterates) For the given x^k , set the diagonal matrix $Q = \text{diag}(q_j)_{j=1}^n$ and the gradient vector $g = (g_1, \dots, g_n)^T$ as follows.

$$\begin{cases} q_j \leftarrow p_j & \text{if } l_j \leq x_j^k \leq u_j, \\ q_j \leftarrow p_j + \alpha & \text{otherwise;} \end{cases}$$

$$\begin{cases} g_j \leftarrow q_j x_j^k + c_j & \text{if } l_j \leq x_j^k \leq u_j, \\ g_j \leftarrow q_j x_j^k + c_j - \alpha l_j & \text{if } x_j^k < l_j, \\ g_j \leftarrow q_j x_j^k + c_j - \alpha u_j & \text{if } x_j^k > u_j, \end{cases}$$

where α is the penalty parameter. Compute the Newton direction Δx^k iteratively according to Procedure 2.6. If $\Delta x^k = 0$, go to Step 2; otherwise, do a line search along Δx^k to find x^{k+1} , update k , and repeat Step 1.

Step 2. (Optimality Test for (NQP)) Update k . Check whether the new iterate is a δ -feasible and ϵ -optimal solution of (NQP). If yes, terminate; if no, set $\alpha = 10\alpha$ and go to Step 1. (The ϵ -optimality can be checked in the same way as the network simplex method, i.e., computing the duality gaps.)

2.5. The piecewise quadratic case. The algorithm and related theories can be extended to piecewise quadratic cases without major changes. The piecewise quadratic case has become increasingly important because many recent algorithms for nonsmooth optimization require the solution of piecewise quadratic subproblem; see [27], [29], and [34].

Suppose that in problem (NQP) the objective function is changed to

$$p(x) = \sum_{j=1}^n p_j(x_j),$$

where for $j = 1, 2, \dots, n$,

$$p_j(x_j) = \begin{cases} \frac{1}{2}p_{j1}x_j^2 + c_{j1}x_j + r_{j1} & \text{if } l_j \equiv a_{j0} \leq x_j \leq a_{j1}, \\ \frac{1}{2}p_{j2}x_j^2 + c_{j2}x_j + r_{j2} & \text{if } a_{j1} \leq x_j \leq a_{j2}, \\ \dots & \\ \frac{1}{2}p_{jk_j}x_j^2 + c_{jk_j}x_j + r_{jk_j} & \text{if } a_{j,k_j-1} \leq x_j \leq a_{j,k_j} \equiv u_j. \end{cases}$$

The numbers $a_{j\bar{j}}, \bar{j} = 1, \dots, k_j - 1$, split the interval $[l_j, u_j]$ into k_j subintervals, where k_j is a positive integer. Each $p_j(x_j)$, in addition, is strictly convex piecewise quadratic and continuously differentiable in $[l_j, u_j]$. Then Theorems 2.1–2.3 are all valid for the new case with little changes in the proofs. On the algorithmic aspects, only the definitions of q_j and g_j need some adjustment as follows.

$$\begin{cases} q_j \leftarrow p_{j\bar{j}} & \text{if } a_{j,\bar{j}-1} \leq x_j^k < a_{j,\bar{j}} \text{ for some } \bar{j}, 1 \leq \bar{j} \leq k_j, \\ q_j \leftarrow p_{j0} + \alpha & \text{if } x_j^k < l_j, \\ q_j \leftarrow p_{jk_j} + \alpha & \text{if } x_j^k \geq u_j; \end{cases}$$

$$\begin{cases} g_j \leftarrow q_j x_j^k + c_{j\bar{j}} & \text{if } a_{j,\bar{j}-1} \leq x_j^k < a_{j,\bar{j}} \text{ for some } \bar{j}, 1 \leq \bar{j} \leq k_j, \\ g_j \leftarrow q_j x_j^k + c_{j0} - \alpha l_j & \text{if } x_j^k < l_j, \\ g_j \leftarrow q_j x_j^k + c_{jk_j} - \alpha u_j & \text{if } x_j^k \geq u_j. \end{cases}$$

Once Q and g are defined, the same procedure as Procedure 2.6 can be used for finding the Newton directions. The line search procedure is also unchanged, only more pieces of the function might be checked in order to find the minimum.

3. Results of computational experiments. Some computational experiments have been designed and conducted so as to make a primary judgment on the applicability of the proposed method. The goals of these experiments are

1. testing the effectiveness of the method and observing its behavior in solving problem (NQP) of various sizes and structures, especially the dependence of computational time upon problem size;
2. investigating the relationship between the accuracy of Newton directions and number of iterations, therefore determining the empirical accuracy requirement of the Newton direction and the solution;
3. comparing the efficiency of the algorithm under cold start and warm start;
4. looking for an acceptable rule to select the penalty parameter.

We use NETGEN (a random network problem generator) [23] to produce problems for our experiments. Problems generated by NETGEN have been long used in testing algorithms for network linear programming. The problems include three classes: transportation (TP), assignment (AS), and transshipment (TS). They are relatively unstructured in the sense that all arcs are randomly produced, the networks are relatively dense, and the problems are not associated with any specific application. Therefore, we think that they are suitable to provide general information on the tested algorithm.

We slightly revised the code of NETGEN so that it can generate quadratic network problems. In order to have strictly convex problems, the quadratic terms all have positive coefficients between 1 and 15 according to the uniform distribution. The linear terms have coefficients between -30 and 30 with uniform distribution. The lower and upper bounds of all x_j s are 0 and 100, respectively. The total supply required by NETGEN is chosen in the same way as the benchmark problems in Klingman, Napier, and Stutz [23].

The experimental code is written in FORTRAN, compiled with optimizing option `-o` and run in the UNIX environment on a SUN workstation. The code contains as a subroutine the NETPLP code of Sun and Tsai [35], which is used to give a warm start (i.e., a good initial solution) for the Newton method by solving a piecewise linear approximation of problem (NQP). NETPLP is a simplex code for separable piecewise linear programs. A switch number is given by the user to switch on/off the warm start.

3.1. The general behavior of the algorithm. The first group of experiments includes 20 problems, all with starting $x_N = 0$ (cold start). They are aimed to test the general applicability of the algorithm and the size of the problems that can be solved in reasonable machine time. The problems include all three types: transportation,

assignment, and transshipment of 5, 5, and 10 each. The smallest problem has 200 nodes and 1300 arcs, and the largest has 400 nodes and 10,000 arcs. All penalty parameters are fixed at $\alpha = 10,000$. The algorithm stops if the infinity norm of the reduced gradient is less than 10^{-4} . In each iteration a Newton direction is obtained if the consecutive iterates in process (15) differ by 10^{-6} in their infinity norm.

Only 17 problems have been solved in about 30 minutes of CPU time as shown in Table 3.1. Among them 11 problems are solved within 10 minutes. In all problems more than 90% of CPU time is spent in computing the Newton directions. We also notice that typically 10–50 Newton directions are computed before the algorithm finds the optimum. This number of iterations is in general less than the numbers of iterations required by other methods reported in the literature in solving problems of similar sizes.

TABLE 3.1
Numerical results using fixed tolerance in computing Newton directions.

#	Problem type	Number of arcs	Number of nodes	Number of iterations	CPU time (seconds)	Optimal objective value
1	TP	1500	200	23	110.66	−676.40
2	TP	2272	250	51	267.88	−1028.84
3	TP	3000	400	23	485.39	−1335.24
4	TP	4500	400	25	865.65	−1454.56
5	TP	5000	400	18	1178.92	−1487.69
6	AS	1319	200	24	91.01	−1235.48
7	AS	2107	200	14	179.54	−1373.84
8	AS	3000	250	18	494.05	−1801.37
9	AS	4023	300	16	594.49	−2200.57
10	AS	5027	400	18	887.20	−2922.42
11	TS	1300	200	16	91.74	6132.28
12	TS	2005	200	18	140.89	−3113.73
13	TS	3150	300	18	477.56	−5733.23
14	TS	4018	300	18	838.65	−6610.58
15	TS	5000	300	16	853.90	−3389.74
16	TS	6088	400	16	1802.68	−6043.49
17	TS	7000	400	14	1722.97	−27950.28
18	TS	8000	400	—	—	—
19	TS	9009	400	—	—	—
20	TS	10000	400	—	—	—

“—” indicates that CPU time in solving the problem is over 2000 seconds.

Tolerance in computing Newton directions = .000001; penalty parameter $\alpha = 10000$.

3.2. On the accuracy of the Newton direction. During the test we observed that several thousand iterations are sometimes necessary to achieve an accuracy of 10^{-6} for only one Newton direction. This is somewhat expected because the iterative method is only linearly convergent in theory. Our computational experiences indicate that the accuracy of the Newton direction is crucial in the last few iterations for the algorithm to converge and to be robust. However, this requirement can be relaxed if the current iterates are not close to the optimum. Therefore, an adjustment on the accuracy of the Newton direction between the “initial iterates” and the “final iterates” may save considerable time. Based on this idea, the second group of tests is conducted.

The same 20 problems with the same initial solutions are tested using a self-adaptive mechanism to control the accuracy of the direction-finding subroutine: if

the last direction was found in less than 100 iterations, then we multiply the tolerance by 0.5; otherwise the same error bound is used. When the bound reduces to 10^{-6} , no further adjustment is made. Initially the tolerance is set at 10^{-3} . This reflects a philosophy of progressively increasing the precision of the Newton directions and balancing the workload among consecutive iterations. The results are shown in Table 3.2.

Comparing Table 3.1 with Table 3.2 shows the efficiency of the dynamic strategy. Although the number of iterations in the main loop sometimes goes higher, roughly two-thirds of the CPU time are saved. All 20 problems are solved in about 20 minutes, while 16 of them are solved within 10 minutes.

TABLE 3.2
Results using adaptive tolerance in computing Newton directions.

#	Problem type	Number of arcs	Number of nodes	Number of iterations	CPU time in all (seconds)	Optimal objective value
1	TP	1500	200	16	40.75	-676.31
2	TP	2272	250	20	85.88	-1028.57
3	TP	3000	400	27	225.91	-1335.54
4	TP	4500	400	16	332.57	-1454.96
5	TP	5000	400	17	408.12	-1487.71
6	AS	1319	200	14	31.06	-1235.53
7	AS	2107	200	16	72.53	-1373.73
8	AS	3000	250	17	197.77	-1801.45
9	AS	4023	300	21	223.39	-2200.69
10	AS	5027	400	24	341.18	-2922.39
11	TS	1300	200	13	23.84	6132.35
12	TS	2005	200	16	42.27	-3113.87
13	TS	3150	300	17	139.30	-5733.31
14	TS	4018	300	16	267.96	-6610.64
15	TS	5000	300	23	319.79	-3389.76
16	TS	6088	400	22	503.21	-6043.44
17	TS	7000	400	19	729.94	-27950.27
18	TS	8000	400	21	903.08	-32825.74
19	TS	9009	400	20	1086.36	-22318.67
20	TS	10000	400	24	1319.34	-37930.82

Adjustable minimum tolerance in computing Newton directions = .000001.
Penalty parameter $\alpha = 10000$.

3.3. Cold start versus warm start. A topic in this computational experiment is to judge how crucial a good starting point is for the Newton algorithm for (NQP). To achieve this goal, we compare test results for three different starting approaches. The first group of problems start from scratch, i.e., from $x_N = 0$ (cold start). The second group starts from the optimal solution obtained by solving a piecewise linearized approximate problem of (NQP), in which each arc cost is approximated by a 10-piece piecewise linear function (mild start). The third group is similar to the second, except using 100-piece functions (warm start). The resulting piecewise linear programs are solved by the code NETPLP mentioned at the beginning of section 3.

The results of mild and warm starts are shown in Table 3.3. (The strategy in section 3.2 was used.) The CPU time includes the time to compute the piecewise linear costs and the time spent by NETPLP. It can be seen that, generally speaking, it is better to use a piecewise linear approximation than not to use it, but there is not too much difference between a cold start and a mild start. For the tested problems

the computational time is reduced more significantly from mild start to warm start than from cold start to mild start. Apparently, the results indicate that, within a practical range, as long as the storage and other conditions permit, it is desirable to use finer piecewise linearizations in order to save computational time.

TABLE 3.3
Numerical results of mild start and warm start.

#	Number of iterations (mild start)	Number of iterations (warm start)	CPU time (mild start)	CPU time (warm start)
1	16	11	37.24	29.39
2	24	14	89.27	64.73
3	28	14	247.31	176.42
4	36	16	356.74	282.67
5	38	18	431.96	377.52
6	13	7	29.65	23.91
7	18	9	68.33	51.63
8	23	10	179.82	132.22
9	32	12	247.26	197.09
10	34	14	371.97	273.92
11	12	9	21.21	10.03
12	18	10	39.54	32.82
13	18	12	149.74	86.35
14	21	15	288.33	167.28
15	23	17	357.47	267.23
16	16	21	576.71	392.53
17	20	19	735.09	432.34
18	18	16	894.55	501.13
19	21	20	1107.80	608.76
20	22	16	1264.32	769.20

Adjustable minimum tolerance in computing Newton directions = .000001.
Penalty parameter $\alpha = 10000$.

3.4. The choice of the penalty parameter. Very little theory has been developed in determining a suitable value for the penalty parameter α . It seems to be a general consensus that if α is small, then the solutions to (NPQP) will tend to be infeasible and, on the other hand, if α is large, some kind of numerical difficulties will occur in solving (NPQP). Since it is quite time-consuming to solve problem (NPQP) repeatedly and since it is not apparent to us that the use of small α may save much time in solving (NPQP), we decide to start with a fairly large α to the extent that no numerical difficulty is caused. Our experience confirms an observation made by Gamble, Conn, and Pulleyblank in [16]; i.e., increasing the problem size has very little effect on the best starting value of α . However, since we are using quadratic rather than linear penalties, it is found that the starting α should be much larger than in the linear case. In fact, the choice of $\alpha = 10^5$ produces acceptable results in all tested problems. The reason why we need large α is that for small violations of the bound constraints the quadratic penalty terms penalize less than linear penalty terms. To have some idea of how the size of α will affect the feasibility of the solutions and the optimal values, we tested a small example as follows.

EXAMPLE 3.1.

$$\begin{aligned} \text{Minimize } & 2x_1^2 + 5x_1 + 3.5x_2^2 + 6x_2 + 2.5x_3^2 + 3x_3 \\ & + 4x_4^2 + 7x_4 + x_5^2 + 6x_5 + 1.5x_6^2 + 9x_6 \end{aligned}$$

$$\begin{aligned}
 \text{Subject to} \quad & x_1 - x_3 + x_4 = 4, \\
 & -x_1 + x_2 - x_6 = 8, \\
 & -x_2 - x_4 + x_5 = -5, \\
 & x_3 - x_5 + x_6 = -7, \\
 & 0 \leq x_i \leq 10 \quad \text{for all } i = 1, 2, \dots, 6.
 \end{aligned}$$

The optimal solution is $x_1 = 0$, $x_2 = 8$, $x_3 = 0$, $x_4 = 4$, $x_5 = 7$, and $x_6 = 0$. The optimal value is 455. Table 3.4 shows the outputs of the test.

TABLE 3.4
Numerical results of the use of different penalty parameters.

#	Penalty α	Number of iterations	Maximum error in variables	Absolute error in objective	Relative error in objective
1	10	3	2.18	322.53	7.1×10^{-1}
2	50	4	0.79	128.01	2.8×10^{-1}
3	100	4	0.091	16.27	3.5×10^{-2}
4	500	5	0.0053	0.95	2.3×10^{-3}
5	1000	5	0.00069	0.14	2.9×10^{-4}
6	5000	5	0.000037	0.0073	1.6×10^{-5}
7	10000	5	0.0000012	0.00024	5.2×10^{-7}

The results seem to indicate that large values of α are necessary for reducing the error in the optimal value rather than for reducing the error in feasibility of the variables.

4. Conclusions and final remarks. In this paper we examine a nonsmooth Newton method applied to network quadratic problems and report preliminary computational results. We derive some theoretical results on the convergence of the algorithm. We also suggest and test some heuristics for reducing the workload in computing the exact Newton directions.

The proposed algorithm can be applied to solve separable network programs with convex piecewise quadratic costs without major changes. The importance of convex piecewise quadratic programs has recently emerged because some nonsmooth optimization algorithms use convex piecewise quadratic programs as subproblems, e.g., the successive quadratic programming approach in [27] and the trust region method in [29]. In addition, some stochastic transshipment problems can be directly formulated as convex separable piecewise quadratic network programs [36].

The ideas of penalty methods have been shown to be effective in our computational test for solving randomly generated unstructured network problems of up to 400 nodes. This extends a similar observation made in [16] to the quadratic case. For the given range of coefficients in our experiments the penalty parameter could be set as large as 10^5 to obtain acceptable near-optimal solutions.

Our results offer an example for the practicality of the nonsmooth Newton method presented in [28]. The method yields a small number of Newton iterations. It is found that the warm start can help save additional CPU time. However, only relatively fine piecewise linearization can result in a relatively significant change in the CPU time.

A serious disadvantage of this algorithm is the slow convergence in computing the Newton direction. Improvements in this direction will be essential to make the algorithm truly practical in solving general nonsmooth network problems. More investigations on the convergence of iterative methods need to be conducted in the context

of network programming. This may in turn require more developments in algorithms for nonsmooth optimization problems.

We may consider an extension of the proposed algorithm to nonconvex network problems. There are other available nonsmooth optimization techniques which may be applied to the network case. The trust region method introduced in [29] is apparently interesting because it is applicable to nonconvex cases.

Acknowledgments. We are grateful to the associate editor and three anonymous referees whose comments greatly helped to improve the manuscript.

REFERENCES

- [1] D. P. BERTSEKAS, P. A. HOSEIN, AND P. TSENG, *Relaxation methods for network flow problems with convex arc costs*, SIAM J. Control Optim., 25 (1987), pp. 1219–1243.
- [2] N. BOLAND, C. J. GOH, AND A. I. MEES, *An algorithm for solving quadratic network flow problems*, Appl. Math. Lett., 4 (1991), pp. 61–64.
- [3] P. M. CAMERINI, G. GALBIATI, AND F. MAFFIOLI, *Algorithms for finding optimum trees: Description, use and evaluation*, Ann. Oper. Res., 13 (1988), pp. 265–397.
- [4] H.-G. CHEN, *Forward-Backward Splitting Techniques: Theory and Applications*, Ph.D. dissertation, Department of Applied Mathematics, University of Washington, Seattle, 1994.
- [5] H.-G. CHEN AND R. T. ROCKAFELLAR, *Convergence rates in forward-backward splitting*, SIAM J. Optim., 7 (1997), pp. 421–444.
- [6] V. CHVATAL, *Linear Programming*, Freeman & Co. New York, 1983.
- [7] F. H. CLARKE, *Optimization and Nonsmooth Analysis*, John Wiley and Sons, New York, 1983.
- [8] M. COLLINS, L. COOPER, R. HELGASON, J. KENNINGTON, AND L. LEBLANC, *Solving the pipe network analysis problem using optimization techniques*, Management Sciences, 24 (1978), pp. 747–760.
- [9] A. R. CONN, *Constrained optimization using a nondifferentiable penalty function*, SIAM J. Numer. Anal., 10 (1976), pp. 760–784.
- [10] R. S. DEMBO, *A primal truncated Newton algorithm with application to nonlinear network optimization*, Math. Programming, 31 (1987), pp. 43–72.
- [11] R. S. DEMBO AND J. G. KLINCEWICZ, *A scaled reduced gradient algorithm for network flow problems with convex separable costs*, Math. Programming, 15 (1981), pp. 125–147.
- [12] R. S. DEMBO, J. M. MULVEY, AND S. A. ZENIOS, *Large-scale nonlinear network models and their application*, Oper. Res., 37 (1989), pp. 353–372.
- [13] J. B. DENNIS, *Mathematical Programming and Electrical Networks*, John Wiley and Sons, New York, 1959.
- [14] M. FRANK AND P. WOLFE, *An algorithm for quadratic programming*, Naval Res. Logist., 33 (1956), pp. 95–110.
- [15] D. GABAY, *Applications of the method of multipliers to variational inequalities*, in Augmented Lagrangian Methods: Applications to the Solution of Boundary Value Problems, M. Fortin and G. Glowinski, eds., North-Holland, Amsterdam, 1983.
- [16] A. B. GAMBLE, A. R. CONN, AND W. R. PULLEYBLANK, *A network penalty problem*, Math. Programming, 50 (1991), pp. 53–73.
- [17] A. A. GOLDSTEIN, *Constructive Real Analysis*, Harper and Row, New York, 1967.
- [18] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, MD, 1983.
- [19] W. H. HAGER AND D. W. HEARN, *Application of the dual active set algorithm to quadratic network optimization*, Comput. Optim. Appl., 1 (1993), pp. 349–373.
- [20] M. IRI, *Flows, Transportation and Scheduling*, Academic Press, New York, 1969.
- [21] J. L. KENNINGTON AND R. V. HELGASON, *Algorithms for Network Programming*, John Wiley & Sons, New York, 1980.
- [22] J. G. KLINCEWICZ, *A Newton method for convex separable network flow problems*, Networks, 13 (1983), pp. 427–442.
- [23] D. KLINGMAN, A. NAPIER, AND J. STUTZ, *NETGEN: A program for generating large scale capacitated assignment, transportation, and minimum cost flow network problems*, Management Sciences, 20 (1974), pp. 814–821.
- [24] H. MARKOWITZ, *Portfolio Selection*, John Wiley and Sons, New York, 1959.
- [25] R. MIFFLIN, *An algorithm for constrained optimization with semismooth functions*, Math. Oper. Res., 2 (1977), pp. 191–207.

- [26] J. M. MULVEY, *Nonlinear network models in finance*, Adv. Math. Program. Finan. Plann., 1 (1987), pp. 253–271.
- [27] J.-S. PANG, S.-P. HAN, AND N. RANGARAJ, *Minimization of locally Lipschitzian functions*, SIAM J. Optim., 1 (1991), pp. 57–82.
- [28] L. QI AND J. SUN, *A nonsmooth version of Newton's method*, Math. Programming, 58 (1993), pp. 353–367.
- [29] L. QI AND J. SUN, *A trust region method for minimization of locally Lipschitzian functions*, Math. Programming, 66 (1994), pp. 25–43.
- [30] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
- [31] R. T. ROCKAFELLAR, *Network Flows and Monotropic Optimization*, John Wiley and Sons, New York, 1984.
- [32] J. SUN, *A study on monotropic piecewise quadratic programming*, in Recent Developments in Mathematical Programming, H. I. Kumar, ed., Gordon and Breach Science Publishers, Melbourne, Australia, 1991, pp. 213–236.
- [33] J. SUN, *On piecewise quadratic Newton and trust region problems*, Math. Programming, 76 (1997), pp. 451–467.
- [34] J. SUN AND L. QI, *An interior point algorithm of $O(\sqrt{m}|\ln \epsilon|)$ iterations for C^1 -convex programming*, Math. Programming, 57 (1992), pp. 239–257.
- [35] J. SUN AND K. TSAI, *A simplex method and its implementation for network piecewise linear programming*, Asia-Pacific J. Oper. Res., 14 (1997), pp. 55–68.
- [36] J. SUN, K. TSAI, AND L. QI, *A simplex method for network programs with convex separable piecewise linear costs and its application to stochastic transshipment problems*, in Network Optimization Problems: Algorithms, Applications and Complexity, D. Z. Du and P. M. Pardalos, eds., World Scientific Publishing Co. London, UK, 1993, pp. 281–300.

PRIMAL-DUAL INTERIOR-POINT METHODS FOR SEMIDEFINITE PROGRAMMING: CONVERGENCE RATES, STABILITY AND NUMERICAL RESULTS*

FARID ALIZADEH[†], JEAN-PIERRE A. HAEBERLY[‡], AND MICHAEL L. OVERTON[§]

Abstract. Primal-dual interior-point path-following methods for semidefinite programming are considered. Several variants are discussed, based on Newton's method applied to three equations: primal feasibility, dual feasibility, and some form of centering condition. The focus is on three such algorithms, called the XZ, XZ+ZX, and Q methods. For the XZ+ZX and Q algorithms, the Newton system is well defined and its Jacobian is nonsingular at the solution, under nondegeneracy assumptions. The associated Schur complement matrix has an unbounded condition number on the central path under the nondegeneracy assumptions and an additional rank assumption. Practical aspects are discussed, including Mehrotra predictor-corrector variants and issues of numerical stability. Compared to the other methods considered, the XZ+ZX method is more robust with respect to its ability to step close to the boundary, converges more rapidly, and achieves higher accuracy.

Key words. semidefinite programming, eigenvalue optimization, interior-point method, convex programming

AMS subject classifications. 65K10, 90C25

PII. S1052623496304700

1. Introduction. Let $\mathcal{S}^n$ denote the vector space of real symmetric $n \times n$ matrices. Denote the dimension of this space by

$$(1.1) \quad n^{\overline{2}} = \frac{n(n+1)}{2}.$$

The standard inner product on $\mathcal{S}^n$ is

$$A \bullet B = \operatorname{tr} AB = \sum_{i,j} A_{ij} B_{ij}.$$

By $X \succeq 0$ ($X \succ 0$), where $X \in \mathcal{S}^n$, we mean that X is positive semidefinite (positive definite).

Consider the semidefinite program (SDP)

$$(1.2) \quad \begin{array}{ll} \min_{X \in \mathcal{S}^n} & C \bullet X \\ \text{s.t.} & A_k \bullet X = b_k, \quad k = 1, \dots, m, \\ & X \succeq 0, \end{array}$$

*Received by the editors June 4, 1996; accepted for publication (in revised form) May 27, 1997; published electronically July 20, 1998.

<http://www.siam.org/journals/siopt/8-3/30470.html>

[†]RUTCOR, Rutgers University, New Brunswick, NJ 08854 (alizadeh@rutcor.rutgers.edu). The research of this author was supported in part by National Science Foundation grant CCR-9501941 and Office of Naval Research contract N00014-96-1-0704.

[‡]Mathematics Department, Fordham University, Bronx, NY 10458 (haeberly@murray.fordham.edu). The research of this author was supported in part by National Science Foundation grant CCR-9401119.

[§]Computer Science Department, Courant Institute of Mathematical Sciences, New York University, New York, NY 10012 (overton@cs.nyu.edu). The research of this author was supported in part by National Science Foundation grant CCR-9625955.

where $b \in \mathbb{R}^m$, $C \in \mathcal{S}^n$, and $A_k \in \mathcal{S}^n$, $k = 1, \dots, m$. The dual SDP is

$$(1.3) \quad \begin{array}{ll} \max_{y \in \mathbb{R}^m, Z \in \mathcal{S}^n} & b^T y \\ \text{s.t.} & \sum_{k=1}^m y_k A_k + Z = C, \\ & Z \succeq 0. \end{array}$$

The following assumptions hold throughout the paper.

Assumption 1. There exists a primal feasible point $X \succ 0$, and a dual feasible point (y, Z) with $Z \succ 0$.

Assumption 2. The matrices A_k , $k = 1, \dots, m$, are linearly independent; i.e., they span an m -dimensional linear space in $\mathcal{S}^n$.

The *central path* consists of points $(X^\mu, y^\mu, Z^\mu) \in \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n$ satisfying the primal and dual feasibility constraints as well as the centering condition

$$(1.4) \quad X^\mu Z^\mu = \mu I$$

for some $\mu \in \mathbb{R}$, $\mu > 0$. It is well known [11, 7] that under Assumptions 1 and 2 (X^μ, y^μ, Z^μ) exists and is unique for all $\mu > 0$, and that

$$(1.5) \quad (X, y, Z) = \lim_{\mu \rightarrow 0} (X^\mu, y^\mu, Z^\mu)$$

exists and solves the primal and dual SDPs. Furthermore, because X^μ and Z^μ commute, there exists an orthogonal matrix Q^μ such that

$$(1.6) \quad X^\mu = Q^\mu \text{Diag}(\lambda_1^\mu, \dots, \lambda_n^\mu) (Q^\mu)^T, \quad Z^\mu = Q^\mu \text{Diag}(\omega_1^\mu, \dots, \omega_n^\mu) (Q^\mu)^T,$$

where the λ_i^μ and ω_i^μ , respectively the eigenvalues of X^μ and Z^μ , satisfy

$$(1.7) \quad \lambda_i^\mu \omega_i^\mu = \mu, \quad i = 1, \dots, n.$$

Without loss of generality, assume that

$$(1.8) \quad \lambda_1^\mu \geq \dots \geq \lambda_n^\mu \quad \text{and} \quad \omega_1^\mu \leq \dots \leq \omega_n^\mu.$$

As $\mu \rightarrow 0$, the centering condition (1.4) reduces to the complementarity condition $XZ = 0$, implying that

$$(1.9) \quad X = Q \text{Diag}(\lambda_1, \dots, \lambda_n) Q^T, \quad Z = Q \text{Diag}(\omega_1, \dots, \omega_n) Q^T$$

for some orthogonal matrix Q , with the eigenvalue complementarity condition $\lambda_i \omega_i = 0, i = 1, \dots, n$. Observe that λ_i and ω_i are the limits of λ_i^μ and ω_i^μ as $\mu \rightarrow 0$, and Q may be taken to be a limit point (not necessarily unique) of the set $\{Q^\mu : \mu > 0\}$. We have

$$(1.10) \quad \lambda_1 \geq \dots \geq \lambda_n \quad \text{and} \quad \omega_1 \leq \dots \leq \omega_n.$$

Interior-point methods for semidefinite programming were originally introduced by [11, 4]. Early papers on primal-dual methods include [17] and [6]. A preliminary version of the present work appeared as [2]. Convergence analysis of primal-dual path-following methods for SDP appeared first in [7, 13, 12]. We are primarily concerned with four methods, which we call the XZ, XZ+ZX, Nesterov–Todd (NT), and Q methods. The XZ method first appeared in [6, 7]. The XZ+ZX method was introduced in

[2] and was recently analyzed in [8, 9]. The NT method was given in [13, 12] and its implementation was recently discussed in [16]. The Q method originally appeared in [1]. Many other papers on semidefinite programming have recently been announced.

The paper is organized as follows. In section 2 we introduce several algorithms in a common framework based on Newton's method, focusing on the XZ and XZ+ZX variants. In section 3 we study the Jacobian of the Newton system for the various methods under nondegeneracy assumptions, and discuss implications for local convergence rates. In section 4 we consider the conditioning of the Schur complement matrix on the central path, again under nondegeneracy assumptions. This leads to the issue of numerical stability, discussed in section 5. We introduce the Q method in section 6. In section 7, we present computational results.

Our main focus is on the nondegenerate case; this assumption (defined in section 3) implies unique primal and dual solutions. We take the view that it is important to understand how methods behave on nondegenerate problems. This does not discount the significance of degenerate problems that may arise in applications, as is common in linear programming (LP).

In practice, many semidefinite programs are block diagonal. Everything in this paper extends easily to the block-diagonal case. Note that LP is the special case where all block sizes are one.

A word about notation: we use the symbols X , y , and Z to mean several things. Depending on the context, they may refer to the variables of the SDP, the iterates generated by a method, or a solution of the SDP.

2. The methods in a general framework. We consider only primal-dual interior-point path-following methods, generating a sequence of iterates approximating the central path and converging to the primal and dual solutions. See [18] for a detailed discussion of such methods for LP. In LP, the basic iterative step can be readily derived using Newton's method. For SDP, points on the central path satisfy the nonlinear equation

$$(2.1) \quad \begin{bmatrix} \sum_{k=1}^m y_k A_k + Z - C \\ A_1 \bullet X - b_1 \\ \vdots \\ A_m \bullet X - b_m \\ XZ - \mu I \end{bmatrix} = 0.$$

However, the matrix XZ is not symmetric in general. Consequently, the domain and range of the function defined by the left-hand side of (2.1) are not the same spaces, and Newton's method is not directly applicable. For LP, on the other hand, the standard primal-dual interior-point method is obtained by applying Newton's method to (2.1). In this case, X and Z are diagonal, and XZ is also diagonal, so the domain and range of (2.1) reduce to $\mathbb{R}^{2n+m}$.

A key question in formulating primal-dual interior-point methods for SDP is therefore, how should one appropriately formulate Newton's method? We consider here two possibilities. Other choices are discussed at the end of this section.

The XZ method. Use the centering condition (1.4) directly and view the left-hand side of (2.1) as a function whose domain and range are both $\mathcal{U} = \mathbb{R}^{n \times n} \times \mathbb{R}^m \times \mathbb{R}^{n \times n}$. Then Newton's method is well defined, though the iterates are not symmetric matrices. (Actually, only the X iterates are not symmetric, since the dual feasibility equation forces Z to be symmetric.) The

X iterates can then be explicitly symmetrized before continuing with the next iteration. Consequently, this method is not strictly a Newton method. A different iteration is obtained by using $ZX = \mu I$ instead of (1.4).

The $XZ + ZX$ method. Rewrite (1.4) in the symmetric form

$$(2.2) \quad XZ + ZX = 2\mu I.$$

Substituting (2.2) for (1.4) in (2.1) gives a mapping with domain and range both given by $\mathcal{V} = \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n$. Application of Newton's method to (2.2) leads to symmetric matrix iterates X and Z .

We observe that (1.4) and (2.2) are equivalent when $X \succeq 0$ (or $Z \succeq 0$). That (1.4) implies (2.2) is immediate. That the converse holds for $X \succeq 0$ is seen by using $X = Q\Lambda Q^T$ to reduce (2.2) to $\Lambda(Q^T Z Q) + (Q^T Z Q)\Lambda = 2\mu I$, with Λ diagonal and nonnegative and $Q^T Q = I$. The entries on the left-hand side are $(\lambda_i + \lambda_j)(QZQ^T)_{ij}$, and so, since the off-diagonal entries must be zero, either $\lambda_i = \lambda_j = 0$ or $(QZQ^T)_{ij} = 0$ when $i \neq j$. Thus, $\Lambda(Q^T Z Q)$ is diagonal, and (1.4) holds.

We now examine the steps defined by these methods in more detail. The Newton step for the XZ method satisfies the linear equation

$$(2.3) \quad X \Delta Z + \Delta X Z = \mu I - XZ.$$

Let nvec map $\mathbb{R}^{n \times n}$ to $\mathbb{R}^{n^2}$, stacking the columns of a matrix in a vector. Then we may rewrite (2.3) in the form

$$(2.4) \quad (I \otimes X) \text{nvec}(\Delta Z) + (Z \otimes I) \text{nvec}(\Delta X) = \text{nvec}(\mu I - XZ),$$

where $\otimes$ denotes the standard Kronecker product (see the Appendix, equation (7.3)).

To discuss the XZ+ZX method, we introduce a *symmetric* version of the Kronecker product. The Newton correction for (2.2) satisfies the linear equation

$$(2.5) \quad X \Delta Z + \Delta Z X + \Delta X Z + Z \Delta X = 2\mu I - XZ - ZX,$$

where ΔX and ΔZ are symmetric. Let svec be an isometry identifying $\mathcal{S}^n$ with $\mathbb{R}^{n^2}$, so that $K \bullet L = \text{svec}(K)^T \text{svec}(L)$ for all $K, L \in \mathcal{S}^n$ (see Appendix). Then (2.5) can be written as

$$(2.6) \quad (Z \circledast I) \text{svec}(\Delta X) + (X \circledast I) \text{svec}(\Delta Z) = \text{svec}\left(\mu I - \frac{1}{2}(XZ + ZX)\right),$$

where $\circledast$ denotes the symmetric Kronecker product defined in the Appendix (see (7.6)).

We shall now describe both methods in a common framework. Let vec denote either nvec or svec , depending on the context. Specifically, vec will mean nvec in the case of the XZ method and svec otherwise. The inverse of vec is denoted by mat . We shall use lower case letters x and z to denote $\text{vec } X$ and $\text{vec } Z$ respectively, and we shall use Δx and Δz interchangeably with $\text{vec } \Delta X$ and $\text{vec } \Delta Z$, to be defined shortly.

Let

$$(2.7) \quad \mathbf{A} = \begin{bmatrix} (\text{vec } A_1)^T \\ \vdots \\ (\text{vec } A_m)^T \end{bmatrix}$$

and define

$$r_p = b - \mathbf{A}x, \quad R_d = C - Z - \text{mat } \mathbf{A}^T y,$$

and

$$(2.8) \quad R_c = \left\{ \begin{array}{ll} \mu I - XZ & \text{XZ method} \\ \mu I - \frac{1}{2}(XZ + ZX) & \text{XZ + ZX method} \end{array} \right\}$$

with

$$r_d = \text{vec } R_d, \quad r_c = \text{vec } R_c.$$

Let

$$(2.9) \quad G(x, y, z) = \begin{bmatrix} -r_d \\ -r_p \\ -r_c \end{bmatrix}.$$

Note that G maps $\mathcal{U}$ to $\mathcal{U}$ in the case of the XZ method and $\mathcal{V}$ to $\mathcal{V}$ otherwise. Application of one step of Newton's method to $G(x, y, z) = 0$ gives the linear system

$$(2.10) \quad \begin{bmatrix} 0 & \mathbf{A}^T & \mathbf{I} \\ \mathbf{A} & 0 & 0 \\ \mathbf{E} & 0 & \mathbf{F} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} r_d \\ r_p \\ r_c \end{bmatrix}.$$

Here

$$\mathbf{E} = \left\{ \begin{array}{ll} Z \otimes I & \text{XZ method} \\ Z \circledast I & \text{XZ + ZX method} \end{array} \right\}$$

and

$$\mathbf{F} = \left\{ \begin{array}{ll} I \otimes X & \text{XZ method} \\ X \circledast I & \text{XZ + ZX method} \end{array} \right\}$$

and $\mathbf{I}$ is the identity matrix of appropriate dimension ($I \otimes I$ for the XZ method and $I \circledast I$ for the XZ+ZX method). We denote the Jacobian matrix on the left-hand side of (2.10) by $\mathbf{J}$.

Applying block Gauss elimination, (2.10) reduces to the system

$$(2.11) \quad \begin{bmatrix} -\mathbf{F}^{-1}\mathbf{E} & \mathbf{A}^T \\ \mathbf{A} & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} r_d - \mathbf{F}^{-1}r_c \\ r_p \end{bmatrix}.$$

A second step of block Gauss elimination gives

$$(2.12) \quad \mathbf{M}\Delta y = r_p + \mathbf{A}\mathbf{E}^{-1}(\mathbf{F}r_d - r_c),$$

$$(2.13) \quad \Delta x = -\mathbf{E}^{-1}(\mathbf{F}(r_d - \mathbf{A}^T \Delta y) - r_c),$$

and, from the dual feasibility equation,

$$(2.14) \quad \Delta z = r_d - \mathbf{A}^T \Delta y,$$

where

$$(2.15) \quad \mathbf{M} = \mathbf{A}\mathbf{E}^{-1}\mathbf{F}\mathbf{A}^T.$$

We call $\mathbf{M}$ the *Schur complement* matrix. The main computational work is the formation and factorization of $\mathbf{M}$. The k th column of the matrix $\mathbf{E}^{-1}\mathbf{F}\mathbf{A}^T$ is

$$\left\{ \begin{array}{ll} \text{nvec}(XA_kZ^{-1}) & \text{XZ method} \\ \text{svec}(G_k) & \text{XZ + ZX method} \end{array} \right\},$$

where G_k is the solution of the Lyapunov equation (see Appendix)

$$(2.16) \quad ZG_k + G_kZ = XA_k + A_kX.$$

Formation of $\mathbf{M}$ thus requires $O(mn^3 + m^2n^2)$ work, involving a Cholesky factorization of Z , in the case of the XZ method, and an eigenvalue factorization of Z , in the case of the XZ+ZX method (see Appendix). Neglecting sparsity considerations, the additional cost of the eigenvalue factorization is negligible in comparison to the other operations required to form $\mathbf{M}$.

It is clear that, as long as $X \succ 0$ and $Z \succ 0$, nonsingularity of the Jacobian matrix $\mathbf{J}$ is equivalent to nonsingularity of the Schur complement $\mathbf{M}$. In the case of the XZ method, $\mathbf{M}$ is symmetric and positive definite. In the case of the XZ+ZX method, $\mathbf{M}$ is not symmetric, but can be shown to be nonsingular if $XZ + ZX \succ 0$ [15]. An alternative condition guaranteeing the nonsingularity of $\mathbf{M}$ is given in [10]. Equation (2.12) is solved by using a Cholesky factorization of $\mathbf{M}$ in the case of the XZ method and an LU factorization of $\mathbf{M}$ in the case of the XZ+ZX method.

For the XZ+ZX method, the multiplications by $\mathbf{E}^{-1}$ in (2.12) and (2.13) require the solution of Lyapunov equations, using the eigenvalues of Z already computed to form $\mathbf{M}$.

Both methods are then described by the following iteration.

BASIC ITERATION.

1. Choose $0 \leq \sigma < 1$ and define

$$(2.17) \quad \mu = \sigma \frac{X \bullet Z}{n}.$$

2. Determine ΔX , Δy , ΔZ from (2.10), equivalently (2.12)–(2.14).
3. In the case of the XZ method, replace ΔX by $\frac{1}{2}(\Delta X + \Delta X^T)$.
4. Choose steplengths α, β and update the iterates by

$$\begin{aligned} X &\leftarrow X + \alpha \Delta X, \\ y &\leftarrow y + \beta \Delta y, \\ Z &\leftarrow Z + \beta \Delta Z. \end{aligned}$$

Rules for defining σ will be discussed later. A simple steplength rule is given by choosing a parameter τ , $0 < \tau < 1$, and defining

$$(2.18) \quad \alpha = \min(1, \tau \hat{\alpha}), \quad \hat{\alpha} = \sup\{\bar{\alpha} : X + \bar{\alpha} \Delta X \succeq 0\}$$

and

$$(2.19) \quad \beta = \min(1, \tau \hat{\beta}), \quad \hat{\beta} = \sup\{\bar{\beta} : Z + \bar{\beta} \Delta Z \succeq 0\}.$$

Note that, except in the case $\Delta X \succeq 0$, we have $0 < \hat{\alpha} < \infty$ with

$$\hat{\alpha}^{-1} = \lambda_{\max}(-L^{-1}\Delta XL^{-T}),$$

where $\lambda_{\max}$ means the largest eigenvalue and L is the Cholesky factor of X , i.e., $X = LL^T$.

Other methods can also be defined in the same framework; two of these are discussed below. See [19] for a class of methods that includes the XZ+ZX method, and [7, 15] for another class that includes all those discussed here except the XZ+ZX method.

The X^{-1} method. Replace R_c in (2.9) by $R_c = \mu X^{-1} - Z$, so $\mathbf{E} = \mu X^{-1} \otimes X^{-1}$, $\mathbf{F} = I \otimes I$. A similar method can be defined with $R_c = \mu Z^{-1} - X$. In fact, the method given by [17] is based on a combination of these two steps.

The NT method. Use $R_c = \mu X^{-1} - Z$, $\mathbf{E} = W^{-1} \otimes W^{-1}$, $\mathbf{F} = I \otimes I$, where $W = X^{1/2}(X^{1/2}ZX^{1/2})^{-1/2}X^{1/2}$. This form does not actually arise from applying Newton's method to (2.9). However, see [16] for a Newton interpretation of this method.

As long as $X \succ 0$ and $Z \succ 0$, $\mathbf{E}^{-1}\mathbf{F}$ is symmetric and positive definite for both these methods. However, in both cases, the function to which Newton's method is applied fails to exist at a solution. We call an algorithm a Newton method if $(\Delta X, \Delta y, \Delta Z)$ is derived by applying Newton's method to a function that is well defined for all $X \succeq 0$, $Z \succeq 0$. Under this definition, of the four variants defined so far, only XZ+ZX is a Newton method for SDP.

In the special case of LP (i.e., a block-diagonal SDP with block sizes all one), the XZ, XZ+ZX, and NT methods coincide, giving the XZ method for LP, which is a Newton method.

In order to understand the asymptotic behavior of Newton's method, it is important to analyze the Jacobian at the solution itself. This is done in the next section.

3. The Jacobian at the solution. In this section we study the Jacobian of the function G , appearing on the left-hand side of (2.10), under nondegeneracy assumptions. To do this, we use the notions of nondegeneracy that were introduced by the authors in [3].

DEFINITION 1. Let (X, y, Z) solve (1.2), (1.3) with an orthogonal matrix Q satisfying (1.9). Let X have rank r , with positive eigenvalues $\lambda_1, \dots, \lambda_r$, and partition $Q = [Q_1 \ Q_2]$, where the columns of Q_1 are eigenvectors corresponding to $\lambda_1, \dots, \lambda_r$. We say that (X, y, Z) satisfies the strict complementarity and primal and dual nondegeneracy conditions if the following hold:

1. $\text{rank}(Z) = n - r$,
2. the matrices

$$(3.1) \quad \begin{bmatrix} Q_1^T A_k Q_1 & Q_1^T A_k Q_2 \\ Q_2^T A_k Q_1 & 0 \end{bmatrix} \quad \text{for } k = 1, 2, \dots, m$$

are linearly independent in $\mathcal{S}^n$, and

3. the matrices

$$(3.2) \quad Q_1^T A_k Q_1 \quad \text{for } k = 1, 2, \dots, m$$

span the space $\mathcal{S}^r$.

These conditions are well defined even if Q is not unique. The first requirement is the strict complementarity condition. Conditions (3.1), (3.2) are respectively primal

and dual nondegeneracy conditions under the assumption of strict complementarity. They immediately imply the inequalities

$$(3.3) \quad r^2 \leq m \leq r^2 + r(n - r)$$

(recalling the notation (1.1)). They also imply uniqueness of the primal and dual solutions. Furthermore, the conditions are generic properties of SDPs, meaning roughly that they hold with probability one for an optimal solution triple, given random data with feasible solutions. For motivation of these conditions and further details, see [3]. The definitions are easily extended to the block-diagonal case, giving the usual LP nondegeneracy conditions when all blocks have size one.

The strict complementarity condition $\text{rank}(X) = r$, $\text{rank}(Z) = n - r$ implies, using (1.10), that

$$(3.4) \quad \lambda_1 \geq \cdots \geq \lambda_r > \lambda_{r+1} = \cdots = \lambda_n = 0$$

and

$$(3.5) \quad 0 = \omega_1 = \cdots = \omega_r < \omega_{r+1} \leq \cdots \leq \omega_n.$$

Let $B_k = Q^T A_k Q$. From (7.6), we have

$$\text{svec } B_k = (Q^T \otimes Q^T) \text{svec } A_k.$$

Recall the definition (2.7), and define

$$(3.6) \quad \mathbf{B} = \begin{bmatrix} (\text{svec } B_1)^T \\ \vdots \\ (\text{svec } B_m)^T \end{bmatrix},$$

so that

$$\mathbf{A}(Q \otimes Q) = \mathbf{B}.$$

Each column of $\mathbf{B}$ corresponds to an index pair (i, j) , identifying two columns of Q , with $1 \leq i \leq j \leq n$. By choosing the ordering used by the svec operator appropriately, we may write

$$(3.7) \quad \mathbf{B} = [\mathbf{C}_1 \ \mathbf{C}_2 \ \mathbf{C}_3],$$

where $\mathbf{C}_1$ contains r^2 columns corresponding to $1 \leq i \leq j \leq r$, $\mathbf{C}_2$ contains $r(n - r)$ columns corresponding to $1 \leq i \leq r$, $r + 1 \leq j \leq n$, and $\mathbf{C}_3$ consists of $(n - r)^2$ columns corresponding to $r + 1 \leq i \leq j \leq n$. The primal nondegeneracy condition (3.1) holds exactly when the rows of $[\mathbf{C}_1 \ \mathbf{C}_2]$ are linearly independent, i.e., $[\mathbf{C}_1 \ \mathbf{C}_2]$ has rank m . The dual nondegeneracy condition (3.2) holds exactly when $\mathbf{C}_1$ has rank r^2 , i.e., the columns of $\mathbf{C}_1$ are linearly independent. Thus, the conditions (3.1) and (3.2) together imply that it is possible to choose $m - r^2$ columns from $\mathbf{C}_2$ so that, together with *all* the columns of $\mathbf{C}_1$, they form a nonsingular $m \times m$ matrix. In other words, we can choose an ordering for the columns of $\mathbf{C}_2$, and therefore of $\mathbf{B}$, so that

$$(3.8) \quad \mathbf{B} = [\mathbf{B}_1 \ \mathbf{B}_2],$$

where $\mathbf{B}_1 \in \mathbb{R}^{m \times m}$ is nonsingular.

THEOREM 3.1. *Consider an SDP whose solution (X, y, Z) satisfies the strict complementarity and primal and dual nondegeneracy conditions. Let $\mathbf{J}$ be the Jacobian of the function G defining the XZ+ZX method, evaluated at (X, y, Z) . Then $\mathbf{J}$ is nonsingular.*

Proof. We have

$$\mathbf{J} = \begin{bmatrix} 0 & \mathbf{A}^T & \mathbf{I} \\ \mathbf{A} & 0 & 0 \\ \mathbf{E} & 0 & \mathbf{F} \end{bmatrix},$$

where $\mathbf{E} = Z \otimes I$ and $\mathbf{F} = X \otimes I$. Let $\mathbf{P} = Q \otimes Q$, and let $\mathbf{S} = \text{Diag}(\mathbf{P}, \mathbf{I}, \mathbf{P})$, so that

$$\mathbf{S}^T \mathbf{J} \mathbf{S} = \begin{bmatrix} 0 & \mathbf{B}^T & \mathbf{I} \\ \mathbf{B} & 0 & 0 \\ \mathbf{\Upsilon} & 0 & \mathbf{\Phi} \end{bmatrix},$$

with $\mathbf{\Phi} = \mathbf{P}^T \mathbf{F} \mathbf{P}$ and $\mathbf{\Upsilon} = \mathbf{P}^T \mathbf{E} \mathbf{P}$. Using Lemma 7.2 (see Appendix) and (1.9), we see that $\mathbf{P}^T \mathbf{P} = \mathbf{I}$, and $\mathbf{\Phi}$ and $\mathbf{\Upsilon}$ are diagonal with entries $\frac{1}{2}(\lambda_i + \lambda_j)$ and $\frac{1}{2}(\omega_i + \omega_j)$, $1 \leq i \leq j \leq n$, respectively. Notice that the diagonal entry of $\mathbf{\Phi}$ corresponding to the index pair (i, j) is zero if and only if $r + 1 \leq i \leq j \leq n$ (because of (3.4)), while the diagonal entry of $\mathbf{\Upsilon}$ corresponding to the pair (i, j) is zero if and only if $1 \leq i \leq j \leq r$ (see (3.5)).

Using the partitioning of $\mathbf{B}$ in (3.8), we have

$$(3.9) \quad \mathbf{S}^T \mathbf{J} \mathbf{S} = \begin{bmatrix} 0 & 0 & \mathbf{B}_1^T & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{B}_2^T & 0 & \mathbf{I} \\ \mathbf{B}_1 & \mathbf{B}_2 & 0 & 0 & 0 \\ \mathbf{\Upsilon}_1 & 0 & 0 & \mathbf{\Phi}_1 & 0 \\ 0 & \mathbf{\Upsilon}_2 & 0 & 0 & \mathbf{\Phi}_2 \end{bmatrix},$$

where $\mathbf{\Upsilon} = \text{Diag}(\mathbf{\Upsilon}_1, \mathbf{\Upsilon}_2)$ and $\mathbf{\Phi} = \text{Diag}(\mathbf{\Phi}_1, \mathbf{\Phi}_2)$. We have $\mathbf{\Phi}_1 \succ 0$, since none of the columns of $\mathbf{C}_3$ are included in $\mathbf{B}_1$, and $\mathbf{\Upsilon}_2 \succ 0$, since all of the columns of $\mathbf{C}_1$ are included in $\mathbf{B}_1$.

Interchanging the first and third rows and the second and last columns of (3.9), we obtain

$$\begin{bmatrix} \mathbf{B}_1 & 0 & 0 & 0 & \mathbf{B}_2 \\ 0 & \mathbf{I} & \mathbf{B}_2^T & 0 & 0 \\ 0 & 0 & \mathbf{B}_1^T & \mathbf{I} & 0 \\ \mathbf{\Upsilon}_1 & 0 & 0 & \mathbf{\Phi}_1 & 0 \\ 0 & \mathbf{\Phi}_2 & 0 & 0 & \mathbf{\Upsilon}_2 \end{bmatrix}.$$

We shall demonstrate the nonsingularity of this matrix using block Gauss elimination. First, subtract $\mathbf{\Upsilon}_1 \mathbf{B}_1^{-1}$ times the first block row from the fourth block row to eliminate $\mathbf{\Upsilon}_1$ from the (4,1) position. This does not otherwise change the lower triangle or the diagonal blocks, but only introduces $-\mathbf{\Upsilon}_1 \mathbf{B}_1^{-1} \mathbf{B}_2$ into the (4,5) position. Second, subtract $\mathbf{\Phi}_2$ times the second block row from the fifth row, eliminating $\mathbf{\Phi}_2$ from the (5,2) position; this introduces $-\mathbf{\Phi}_2 \mathbf{B}_2^T$ into the (5,3) position. This (5,3) block is then eliminated by adding $\mathbf{\Phi}_2 \mathbf{B}_2^T \mathbf{B}_1^{-T}$ times the third row to the fifth row, introducing

$\Phi_2 \mathbf{B}_2^T \mathbf{B}_1^{-T}$ into the (5,4) position, giving

$$\begin{bmatrix} \mathbf{B}_1 & 0 & 0 & 0 & \mathbf{B}_2 \\ 0 & \mathbf{I} & \mathbf{B}_2^T & 0 & 0 \\ 0 & 0 & \mathbf{B}_1^T & \mathbf{I} & 0 \\ 0 & 0 & 0 & \Phi_1 & -\Upsilon_1 \mathbf{H} \\ 0 & 0 & 0 & \Phi_2 \mathbf{H}^T & \Upsilon_2 \end{bmatrix},$$

where $\mathbf{H} = \mathbf{B}_1^{-1} \mathbf{B}_2$. In order to show that this matrix is nonsingular we need only show that the trailing 2×2 block is nonsingular, or equivalently that its positive row scaling

$$\begin{bmatrix} \mathbf{I} & -\Phi_1^{-1} \Upsilon_1 \mathbf{H} \\ \Upsilon_2^{-1} \Phi_2 \mathbf{H}^T & \mathbf{I} \end{bmatrix}$$

is nonsingular. A final step of block Gauss elimination yields a block upper triangular matrix with last diagonal block given by

$$\mathbf{I} + \Upsilon_2^{-1} \Phi_2 \mathbf{H}^T \Phi_1^{-1} \Upsilon_1 \mathbf{H}.$$

This matrix is nonsingular, since it is of the form $\mathbf{I} + \mathbf{N}_1 \mathbf{N}_2$ with $\mathbf{N}_1 \succeq 0$ and $\mathbf{N}_2 \succeq 0$. (The product of two symmetric and positive semidefinite matrices, though nonsymmetric, has real nonnegative eigenvalues.) $\square$

COROLLARY 3.2. *Consider an SDP whose solution (X, y, Z) satisfies the strict complementarity and primal and dual nondegeneracy conditions. Suppose that the XZ+ZX method uses $\sigma = 0$ and $\alpha = \beta = 1$ in the Basic Iteration. Then, there exists $\epsilon > 0$ such that, if the iteration is started at (X_0, y_0, Z_0) , with $\|(X_0, y_0, Z_0) - (X, y, Z)\| < \epsilon$, the iterates converge Q -quadratically to (X, y, Z) .*

The proof of Corollary 3.2 is immediate from the standard convergence theory for Newton's method. It is clear that Corollary 3.2 holds also for less restrictive assumptions on σ, α , and β . See [20] for relevant results for LP. There is no requirement that (X_0, y_0, Z_0) lie in a horn-shaped neighborhood of the central path, or even in the feasible region. Note that the assumptions of Corollary 3.2 do not guarantee positive definite iterates. These are not required to make (2.10) well defined, though the equivalence of (2.10) with (2.12) to (2.14) does not hold if X or Z is singular. In practice, conditions (2.18) and (2.19) ensure positive definite iterates.

A result like Theorem 3.1 does not hold for any of the other methods discussed so far. As already noted, the function to which Newton's method is applied is, in the case of the X^{-1} and NT methods, not defined at an optimal point. For the XZ method, the function G is defined at the solution, but it can be shown that the Jacobian $\mathbf{J}$ is always singular there. More importantly, bearing in mind the symmetrization step, an example can be constructed where $\mathbf{J}$ has a null vector $(\Delta X, \Delta y, \Delta Z)$ with $\Delta X + \Delta X^T \neq 0$.

It is well known that a result like Theorem 3.1 holds for the XZ method for LP, using LP nondegeneracy assumptions.

Nondegeneracy assumptions are not required to obtain superlinear convergence results. This has been known for some years for LP [18] and is the subject of active current research for SDPs. However, such results require that the iterates of a method stay close to the central path. Our point here is that classical Newton theory applies to the XZ+ZX method, under nondegeneracy assumptions, in SDPs just as in LP.

4. Conditioning of the Schur complement matrix. In this section, we study the conditioning of the Schur complement matrix $\mathbf{M}$, introduced in section 2, on the central path. It is important to note that, when started on the central path, all the methods discussed so far generate the same first iterate. On the central path, X and Z commute. Therefore,

$$\mathbf{E}^{-1}\mathbf{F} = \frac{1}{\mu}X \otimes X$$

in all cases except the XZ method for which we have $\mathbf{E}^{-1}\mathbf{F} = \frac{1}{\mu}X \otimes X$. In both cases the Schur complement matrix $\mathbf{M} = \mathbf{A}\mathbf{E}^{-1}\mathbf{F}\mathbf{A}^T$ is the same.

We now analyze the condition number of $\mathbf{M}$ on the central path, as $\mu \rightarrow 0$. We begin by considering its rank in the limit.

THEOREM 4.1. *Assume that (X^μ, y^μ, Z^μ) lies on the central path of an SDP whose solution $(X, y, Z) = \lim_{\mu \rightarrow 0}(X^\mu, y^\mu, Z^\mu)$ satisfies the dual nondegeneracy condition (3.2), with $r = \text{rank}(X)$. Let $\mathbf{M}^\mu$ be the Schur complement matrix defined at (X^μ, y^μ, Z^μ) . Then*

$$\lim_{\mu \rightarrow 0}(\mu \mathbf{M}^\mu)$$

exists and has rank r^2 .

Proof. Clearly,

$$\mu \mathbf{M}^\mu \rightarrow \mathbf{N} = \mathbf{A}(X \otimes X)\mathbf{A}^T,$$

the matrix whose (l, k) element is $\text{tr}(XA_lXA_k)$. Let Q and λ_i satisfy (1.9), and write $\Lambda_1 = \text{Diag}(\lambda_1, \dots, \lambda_r) \succ 0$, with corresponding eigenvectors collected in Q_1 , so that $X = Q_1\Lambda_1Q_1^T$. Let $\mathbf{C}_1$ be the $m \times r^2$ matrix introduced in (3.7), and let $\mathbf{D}_1$ be the $r^2 \times r^2$ diagonal matrix

$$\mathbf{D}_1 = \text{Diag}(\lambda_i\lambda_j), \quad 1 \leq i \leq j \leq r,$$

using consistent orderings for $\mathbf{C}_1$ and $\mathbf{D}_1$. Then

$$\mathbf{N} = \mathbf{C}_1\mathbf{D}_1\mathbf{C}_1^T,$$

since the (l, k) element of the right-hand side is

$$\text{tr}(\Lambda_1Q_1^TA_lQ_1\Lambda_1Q_1^TA_kQ_1) = \text{tr}(XA_lXA_k).$$

Since, by the dual nondegeneracy assumption, $\mathbf{C}_1$ has linearly independent columns, and since $\mathbf{D}_1 \succ 0$, this completes the proof of the theorem. $\square$

Recall that the condition number of a symmetric positive definite matrix is $\kappa_{\max}/\kappa_{\min}$, where $\kappa_{\max}$ and $\kappa_{\min}$ are respectively its largest and smallest eigenvalues.

THEOREM 4.2. *Suppose that the assumptions of Theorem 4.1 hold. Then, if $m > r^2 > 0$, the condition number of $\mathbf{M}^\mu$ (equivalently of $\mu\mathbf{M}^\mu$) is bounded below by a positive constant times $1/\mu$.*

Proof. Let Q^μ , λ_i^μ satisfy (1.6), and let $\mathbf{B}^\mu, \mathbf{C}^\mu$ be the matrices introduced in section 3, evaluated at (X^μ, y^μ, Z^μ) . Using Lemma 7.2 (see Appendix), we have

$$(4.1) \quad \mu \mathbf{M}^\mu = \mathbf{A}(X^\mu \otimes X^\mu)\mathbf{A}^T = \mathbf{B}^\mu\mathbf{D}^\mu(\mathbf{B}^\mu)^T,$$

where $\mathbf{D}^\mu$ is the diagonal $n^{\bar{2}} \times n^{\bar{2}}$ matrix

$$(4.2) \quad \mathbf{D}^\mu = \text{Diag}(\lambda_i^\mu \lambda_j^\mu), \quad 1 \leq i \leq j \leq n.$$

The primal solution rank r defines a splitting

$$\mathbf{D}^\mu = \text{Diag}(\mathbf{D}_1^\mu, \mathbf{D}_2^\mu, \mathbf{D}_3^\mu)$$

consistent with (3.7), so that

$$(4.3) \quad \mu \mathbf{M}^\mu = \mathbf{C}_1^\mu \mathbf{D}_1^\mu (\mathbf{C}_1^\mu)^T + \mathbf{C}_2^\mu \mathbf{D}_2^\mu (\mathbf{C}_2^\mu)^T + \mathbf{C}_3^\mu \mathbf{D}_3^\mu (\mathbf{C}_3^\mu)^T.$$

Here the entries of the diagonal matrices $\mathbf{D}_1^\mu$, $\mathbf{D}_2^\mu$, and $\mathbf{D}_3^\mu$ are $\lambda_i^\mu \lambda_j^\mu$, with the indices $1 \leq i \leq j \leq r$ for $\mathbf{D}_1^\mu$, $1 \leq i \leq r < j \leq n$ for $\mathbf{D}_2^\mu$, and $r+1 \leq i \leq j \leq n$ for $\mathbf{D}_3^\mu$. Although Q^μ and $\mathbf{C}^\mu$ do not generally converge as $\mu \rightarrow 0$, Theorem 4.1 shows that $\mu \mathbf{M}^\mu \rightarrow \mathbf{N} = \mathbf{C}_1 \mathbf{D}_1 \mathbf{C}_1^T$, with rank $r^{\bar{2}}$. By assumption, $m > r^{\bar{2}} > 0$, so the largest eigenvalue of $\mathbf{N}$ is positive and the smallest is zero. The norms of the second and third terms in (4.3) are $O(\mu)$, so the largest and smallest eigenvalues of $\mu \mathbf{M}^\mu$ are, respectively, bounded away from zero and $O(\mu)$. (Here we use the fact that eigenvalues of a symmetric matrix are Lipschitz continuous functions of the matrix entries.) $\square$

Theorem 4.2 is easily extended to the block-diagonal case. When all block sizes are one, the condition on m in its hypothesis cannot hold under the nondegeneracy assumptions. Indeed, it is well known that for LP, under assumptions of nondegeneracy and strict complementarity, the condition number of the Schur complement matrix is bounded independent of μ .

5. Stability. We have seen in the previous section that, for nondegenerate SDPs, the condition number of the Schur complement matrix, evaluated on the central path, is bounded below by a positive constant times $1/\mu$ (ruling out the exceptional cases $r^{\bar{2}} = m$ and $r = 0$). Consequently, we expect that as $\mu \rightarrow 0$, the computation of Δy in (2.12) will become increasingly less accurate. Indeed, in our original implementations we observed numerical instability leading to significant loss of primal feasibility near a solution. Recently, however, Todd, Toh, and Tütüncü [16] found that high accuracy is achievable. The main issue is the choice of formulae for Δy and ΔX . Several mathematically equivalent choices are possible, but these have quite different stability properties.

Formulae for Δy and ΔX are given in (2.12) and (2.13). Both include the term $\mathbf{F}r_d - r_c$. For the XZ+ZX method, this term (in matrix form) is

$$\frac{1}{2} ((X(C - Z - \text{mat } \mathbf{A}^T y) + (C - Z - \text{mat } \mathbf{A}^T y)X) - (2\mu I - XZ - ZX)),$$

which can be rewritten as

$$\frac{1}{2} ((X(C - \text{mat } \mathbf{A}^T y) + (C - \text{mat } \mathbf{A}^T y)X) - 2\mu I).$$

However, using this simplification to modify (2.12) and (2.13) leads to instability and loss of primal feasibility. It is much better to implement (2.12) and (2.13) directly. This is done in the computational experiments reported in section 7.

The same issue applies to the XZ method. However, direct implementation of (2.12) and (2.13) does not give good results for the XZ method. Instead,¹ we use the fact that $\mathbf{E}^{-1}\mathbf{F}$ is symmetric positive definite to write

$$\mathbf{E}^{-1}\mathbf{F} = Z^{-1} \otimes X = \mathbf{G}^T \mathbf{G}, \quad \mathbf{G} = M^{-1} \otimes L^T,$$

where L and M are respectively Cholesky factors of X and Z , i.e.,

$$X = LL^T, \quad Z = MM^T.$$

Noting that the first block in the right-hand side of (2.11) is

$$u = \text{vec } U = \text{vec } (C - \mu X^{-1} - \text{mat } \mathbf{A}^T y),$$

we see that (2.11) is equivalent to

$$(5.1) \quad \begin{bmatrix} -\mathbf{I} & \tilde{\mathbf{A}}^T \\ \tilde{\mathbf{A}} & 0 \end{bmatrix} \begin{bmatrix} \tilde{\Delta}x \\ \Delta y \end{bmatrix} = \begin{bmatrix} \tilde{u} \\ r_p \end{bmatrix},$$

$$\tilde{\Delta}x = \mathbf{G}^{-T} \Delta x = \text{vec } (L^{-1} \Delta X M) = \text{vec } \tilde{\Delta}X,$$

$$\tilde{u} = \mathbf{G}u = \text{vec } (L^T U M^{-T}) = \text{vec } \tilde{U},$$

and

$$(5.2) \quad \tilde{\mathbf{A}} = \mathbf{A}\mathbf{G}^T = \begin{bmatrix} (\text{vec } L^T A_1 M^{-T})^T \\ \vdots \\ (\text{vec } L^T A_m M^{-T})^T \end{bmatrix}.$$

The solution is given by

$$(5.3) \quad (\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T)\Delta y = r_p + \tilde{\mathbf{A}}\tilde{u}$$

(which may be solved with a Cholesky factorization) and

$$(5.4) \quad \Delta X = L\tilde{\Delta}X M^{-1} = L(\text{mat } \tilde{\mathbf{A}}^T \Delta y - \tilde{U})M^{-1}.$$

This last equation can be written in many ways, three of which are

$$(5.5) \quad \Delta X = L(L^T(\text{mat } \mathbf{A}^T \Delta y)M^{-T} - L^T U M^{-T})M^{-1}$$

$$(5.6) \quad = LL^T(\text{mat } \mathbf{A}^T \Delta y)M^{-T}M^{-1} - LL^T U M^{-T}M^{-1}$$

$$(5.7) \quad = LL^T((\text{mat } \mathbf{A}^T \Delta y) - U)M^{-T}M^{-1}.$$

Of these four mathematically equivalent formulae, (5.4) and (5.5) give the highest accuracy, with smallest loss of primal feasibility. We used (5.4) in our computational experiments, with Δy defined by (5.3).

For the NT method, $\mathbf{E}^{-1}\mathbf{F}$ is also symmetric positive definite, so similar considerations apply; see [16].

¹The discussion here is motivated by [16].

6. The Q method. In this section we change direction, deriving an alternative primal-dual interior-point method that generates iterates (X, y, Z) with the property that X and Z commute, i.e., $XZ = ZX$. This is motivated by the fact that this property holds for all points on the central path. Instead of treating the variables X and Z directly, we introduce as variables the eigenvalues of X and Z and their common set of eigenvectors. In other words, the variables consist of an orthogonal matrix Q , diagonal matrices Λ and Ω , and a vector $y \in \mathbb{R}^n$ that must satisfy

$$(6.1) \quad \begin{aligned} Q\Omega Q^T + \sum_{k=1}^m y_k A_k &= C, \\ A_k \bullet (Q\Lambda Q^T) &= b_k, \quad k = 1, \dots, m, \\ \Lambda\Omega &= \mu I. \end{aligned}$$

This defines a map from $\mathcal{O}^n \times \mathbb{R}^{2n+m}$ to $\mathbb{R}^{n^2+n+m}$, where $\mathcal{O}^n$ is the Lie group of orthogonal matrices with determinant one, whose dimension is $n(n-1)/2$. (Since the signs of eigenvectors are arbitrary, it is not a restriction to impose $\det Q = 1$.) The price paid for the diagonalization is the nonlinear appearance of the variable Q in the feasibility equations.

Let $\mathcal{K}^n$ denote the space of $n \times n$ skew-symmetric matrices, and consider the exponential map from $\mathcal{K}^n$ to $\mathcal{O}^n$ defined by

$$\exp(S) = I + S + \frac{1}{2}S^2 + \dots$$

This map is smooth, onto, and, in a neighborhood of 0, also one-to-one. Borrowing a technique used in [14], we derive a form of Newton's method based on parameterizing $\mathcal{O}^n$ near a given point Q by $Q \exp(S)$. Let kvec be an isometry from $\mathcal{K}^n$ to $\mathbb{R}^{n(n-1)/2}$, stacking the upper triangular entries of a skew-symmetric matrix in a vector, with a factor of $\sqrt{2}$ to preserve the inner product. Let us use the convention $s = \text{kvec}(S)$, $\Lambda = \text{Diag}(\lambda)$, and $\Omega = \text{Diag}(\omega)$. Define

$$(6.2) \quad G_Q(\lambda, y, \omega, s) = \begin{bmatrix} \text{vec}(C - Q \exp(S) \Omega \exp(-S) Q^T) - \mathbf{A}^T y \\ b - \mathbf{A} \text{vec}(Q \exp(S) \Lambda \exp(-S) Q^T) \\ \Lambda \Omega e - \mu e \end{bmatrix}.$$

The function G_Q maps $\mathbb{R}^{n^2+n+m}$ to itself. Note that the third component of G_Q has the form familiar from LP.

Given an iterate $(X, y, Z) = (Q\Lambda Q^T, y, Q\Omega Q^T)$, we obtain a new iterate by applying Newton's method to the equation $G_Q = 0$ at the point $(\lambda, y, \omega, 0)$. The Newton step $(\Delta\lambda, \Delta y, \Delta\omega, s)$ is obtained by replacing $\exp(S)$ by $I + S$ and discarding second-order terms. The resulting $(n^2 + n + m) \times (n^2 + n + m)$ linear system is

$$(6.3) \quad \Delta\Omega + S\Omega - \Omega S + \sum_{k=1}^m \Delta y_k B_k = H - \Omega,$$

$$(6.4) \quad B_k \bullet (\Delta\Lambda + S\Lambda - \Lambda S) = b_k - B_k \bullet \Lambda, \quad k = 1, \dots, m,$$

$$(6.5) \quad \Lambda \Delta\Omega + \Omega \Delta\Lambda = \mu I - \Lambda\Omega,$$

where $B_k = Q^T A_k Q$ and $H = Q^T C Q - \sum_{k=1}^m y_k B_k$.

The basic iteration for the Q method is therefore

1. Choose $0 \leq \sigma < 1$ and define

$$\mu = \sigma \frac{\lambda^T \omega}{n}.$$

2. Determine $(\Delta\lambda, \Delta y, \Delta\omega, s)$ from (6.3) to (6.5).
3. Choose steplengths α, β, γ and update the iterates by

$$\begin{aligned} \Lambda &\leftarrow \Lambda + \alpha \Delta\Lambda, \\ y &\leftarrow y + \beta \Delta y, \\ \Omega &\leftarrow \Omega + \beta \Delta\Omega, \\ Q &\leftarrow Q(I + \tfrac{1}{2}\gamma S)(I - \tfrac{1}{2}\gamma S)^{-1}. \end{aligned}$$

A simple steplength rule is $\alpha = \min(1, \tau \hat{\alpha})$, $\beta = \min(1, \tau \hat{\beta})$, and $\gamma = \sqrt{\alpha\beta}$, where $\hat{\alpha}$ and $\hat{\beta}$ are steps to the boundary of the positive orthant. The multiplicative factor updating Q is the Cayley transform, an easily computed orthogonal matrix that approximates the matrix exponential to second order.

The equations defining the Q method can be rewritten as follows. First note that (6.4) can be rewritten as

$$B_k \bullet \Delta\Lambda + \text{tr}((\Lambda B_k - B_k \Lambda)S) = b_k - B_k \bullet \Lambda$$

and write

$$v = \begin{bmatrix} b_1 - B_1 \bullet \Lambda \\ \vdots \\ b_m - B_m \bullet \Lambda \end{bmatrix}.$$

Let $\text{diag}(B_k)$ be the vector consisting of the n diagonal entries of B_k and $\text{offdiag}(B_k)$ be the vector consisting of the $n(n-1)/2$ entries of the upper triangle of B_k , ordered consistently with the ordering chosen for the `kvec` operator. Define

$$\mathbf{L} = [\text{diag}(B_1) \quad \cdots \quad \text{diag}(B_m)]^T,$$

$$\mathbf{R} = [\text{offdiag}(B_1) \quad \cdots \quad \text{offdiag}(B_m)]^T.$$

Let

$$D = \text{Diag}(\lambda_i - \lambda_j), \quad E = \text{Diag}(\omega_i - \omega_j)$$

be diagonal matrices of size $n(n-1)/2$ (corresponding to $1 \leq i < j \leq n$), whose orderings are also consistent with that of the `kvec` operator. Then, writing the diagonal and off-diagonal parts of (6.3) separately, we get the linear system

$$(6.6) \quad \begin{bmatrix} 0 & 0 & \mathbf{L}^T & I \\ 0 & E & \mathbf{R}^T & 0 \\ \mathbf{L} & \mathbf{R}D & 0 & 0 \\ \Omega & 0 & 0 & \Lambda \end{bmatrix} \begin{bmatrix} \Delta\lambda \\ s \\ \Delta y \\ \Delta\omega \end{bmatrix} = \begin{bmatrix} \text{diag}(H - \Omega) \\ \text{offdiag}(H) \\ v \\ \mu e - \Lambda \Omega e \end{bmatrix}.$$

We denote the matrix on the left-hand side of (6.6) by $\mathbf{J}_Q$.

Let (X, y, Z) be a solution of (1.2), (1.3) satisfying (3.4), (3.5). The matrix Q simultaneously diagonalizing X and Z is unique (up to signs of its columns) if and only if

$$(6.7) \quad \lambda_1 > \cdots > \lambda_r > 0 \quad \text{and} \quad 0 < \omega_{r+1} < \cdots < \omega_n.$$

THEOREM 6.1. *Let $(X, y, Z) = (Q\Lambda Q^T, y, Q\Omega Q^T)$ be a solution of (1.2), (1.3) satisfying the strict complementarity and primal and dual nondegeneracy conditions, and also condition (6.7). Then the matrix $\mathbf{J}_Q$, evaluated at the solution, is nonsingular.*

Proof. First note that the assumptions on the eigenvalues imply that the element of the diagonal matrix D corresponding to the index pair (i, j) is zero if and only if $r+1 \leq i < j \leq n$, while the element of the diagonal matrix E corresponding to (i, j) is zero if and only if $1 \leq i < j \leq r$. Let us rewrite $\mathbf{J}_Q$ as

$$\begin{bmatrix} 0 & 0 & 0 & \mathbf{L}_1^T & I & 0 \\ 0 & 0 & 0 & \mathbf{L}_2^T & 0 & I \\ 0 & 0 & E & \mathbf{R}^T & 0 & 0 \\ \mathbf{L}_1 & \mathbf{L}_2 & \mathbf{R}D & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \Lambda_1 & 0 \\ 0 & \Omega_2 & 0 & 0 & 0 & 0 \end{bmatrix},$$

where $\Lambda_1 \succ 0$ and $\Omega_2 \succ 0$. As in the proof of Theorem 3.1, the nondegeneracy assumptions permit us to collect all r columns of $\mathbf{L}_1$ and $m-r$ columns of $\mathbf{R}$ together in a nonsingular $m \times m$ matrix $\mathbf{B}_1$. We collect the remaining $n(n-1)/2 - m + r$ columns of $\mathbf{R}$ in a matrix $\mathbf{R}_2$, and partition $D = \text{Diag}(D_1, D_2)$ and $E = \text{Diag}(E_1, E_2)$ accordingly. Observe that $D_1 \succ 0$ since the columns of $\mathbf{B}_1$ correspond to index pairs (i, j) with $\lambda_i > \lambda_j$. Likewise, $-E_2 \succ 0$ since all columns corresponding to index pairs (i, j) with $\omega_i = \omega_j = 0$ are contained in $\mathbf{B}_1$.

Let $\tilde{D} = \text{Diag}(I, D_1)$ and $\tilde{E} = \text{Diag}(0, E_1)$. Permuting the rows and columns, $\mathbf{J}_Q$ becomes

$$\begin{bmatrix} \tilde{E} & 0 & 0 & \mathbf{B}_1^T & \tilde{I} & 0 \\ 0 & 0 & 0 & \mathbf{L}_2^T & 0 & I \\ 0 & 0 & E_2 & \mathbf{R}_2^T & 0 & 0 \\ \mathbf{B}_1 \tilde{D} & \mathbf{L}_2 & \mathbf{R}_2 D_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \Lambda_1 & 0 \\ 0 & \Omega_2 & 0 & 0 & 0 & 0 \end{bmatrix},$$

where $\tilde{I}$ is an m by r matrix containing r rows of the r by r identity matrix and $m-r$ zero rows. Interchanging the first and fourth rows and the second and last columns, this becomes

$$\begin{bmatrix} \mathbf{B}_1 \tilde{D} & 0 & \mathbf{R}_2 D_2 & 0 & 0 & \mathbf{L}_2 \\ 0 & I & 0 & \mathbf{L}_2^T & 0 & 0 \\ 0 & 0 & E_2 & \mathbf{R}_2^T & 0 & 0 \\ \tilde{E} & 0 & 0 & \mathbf{B}_1^T & \tilde{I} & 0 \\ 0 & 0 & 0 & 0 & \Lambda_1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \Omega_2 \end{bmatrix}.$$

Performing Gauss block elimination on this matrix we see that its nonsingularity is equivalent to the nonsingularity of

$$\mathbf{B}_1^T + \tilde{E} \tilde{D}^{-1} \mathbf{B}_1^{-1} \mathbf{R}_2 D_2 E_2^{-1} \mathbf{R}_2^T.$$

Multiplying on the left by $\mathbf{B}_1^{-T}$ we obtain the matrix

$$I + (\mathbf{B}_1^{-T} \tilde{E} \tilde{D}^{-1} \mathbf{B}_1^{-1})(\mathbf{R}_2 D_2 E_2^{-1} \mathbf{R}_2^T).$$

This is nonsingular since it is of the form $I + N_1 N_2$ with N_1, N_2 symmetric negative semidefinite. $\square$

COROLLARY 6.2. *Consider an SDP whose solution satisfies the strict complementarity and primal and dual nondegeneracy conditions, and also condition (6.7). Suppose that the Q method uses $\sigma = 0$ and $\alpha = \beta = \gamma = 1$. Then, if the method is started with λ, ω, y , and Q initialized sufficiently close to their values at the solution, the iterates converge Q-quadratically to the solution.*

The proof of Corollary 6.2 is more technical than that of Corollary 3.2 and is omitted. It is necessary to establish that quadratic convergence is not impeded by either (a) the use of the Cayley transform to approximate the matrix exponential or (b) the dependence of the definition of G_Q on Q .

As with the other methods, we see how to efficiently implement the Q method by performing block Gauss elimination directly on $\mathbf{J}_Q$ without partitioning the blocks. The first step yields

$$\begin{bmatrix} -\Lambda^{-1}\Omega & 0 & \mathbf{L}^T \\ 0 & D^{-1}E & \mathbf{R}^T \\ \mathbf{L} & \mathbf{R} & 0 \end{bmatrix} \begin{bmatrix} \Delta\lambda \\ \tilde{s} \\ \Delta y \end{bmatrix} = \begin{bmatrix} \text{diag}(H - \mu\Lambda^{-1}) \\ \text{offdiag}(H) \\ v \end{bmatrix},$$

where $\tilde{S} = \text{kvec } \tilde{s}$ is the symmetric matrix defined by

$$\tilde{S}_{ij} = (\lambda_i - \lambda_j)S_{ij}.$$

One more step of block elimination then gives the Schur complement

$$(6.8) \quad \mathbf{M}_Q = [\mathbf{L} \quad \mathbf{R}] \begin{bmatrix} \Lambda\Omega^{-1} & 0 \\ 0 & -DE^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{L}^T \\ \mathbf{R}^T \end{bmatrix}.$$

As in LP, the center factor of the Schur complement is diagonal, with entries

$$\frac{\lambda_i}{\omega_i}, \quad 1 \leq i \leq n, \quad \text{and} \quad \frac{\lambda_i - \lambda_j}{\omega_j - \omega_i}, \quad 1 \leq i < j \leq n.$$

Of course, the $\mathbf{L}$ and $\mathbf{R}$ blocks are not independent of the iteration count.

The Q method does not require computing eigenvalues. The variables Q, λ , and ω are all updated using rational operations. This is in contrast with the XZ+ZX method which requires the computation of eigenvalues in two places: the formation of the Schur complement matrix $\mathbf{M}$ (to solve the Lyapunov equations) and in the step-length computation (to find the step to the boundary). Finally, note that the Schur complement matrix is symmetric for the Q method, but not for the XZ+ZX method.

When evaluated on the central path, the Schur complement matrix $\mathbf{M}_Q^\mu$ for the Q method is equal to the Schur complement matrix $\mathbf{M}^\mu$ for the XZ and XZ+ZX methods, assuming that (6.7) holds. To see this, let $\mathbf{L}^\mu, \mathbf{R}^\mu, D^\mu, E^\mu$ denote the matrices $\mathbf{L}, \mathbf{R}, D, E$ evaluated on the central path. We have $\Lambda^\mu(\Omega^\mu)^{-1} = \text{Diag}(\lambda_i^\mu/\omega_i^\mu) = \frac{1}{\mu}\text{Diag}((\lambda_i^\mu)^2)$ and

$$-D^\mu(E^\mu)^{-1} = \text{Diag}\left(\frac{\lambda_i^\mu - \lambda_j^\mu}{\omega_j^\mu - \omega_i^\mu}\right) = \frac{1}{\mu}\text{Diag}(\lambda_i^\mu \lambda_j^\mu).$$

Thus,

$$\mathbf{M}_Q = \frac{1}{\mu} \mathbf{B}^\mu \mathbf{D}^\mu (\mathbf{B}^\mu)^T = \mathbf{M},$$

using (4.1) and (4.2).

Although the Q method has some attractive features, it is, at present, not a practical alternative to the other algorithms. When initialized far from the solution, convergence is generally not obtained. However, the quadratic local convergence established here is observed in practice.

7. Computational results. In this section we report on the results of some extensive numerical experiments. We start by discussing some important implementation details.

Mehrotra's predictor-corrector (PC) rule is a well-known technique in LP [18]. It can easily be extended to the XZ and XZ+ZX methods, as follows.

XZ and XZ+ZX methods with Mehrotra predictor-corrector rule.

1. Determine ΔX , Δy , ΔZ from (2.10) using $\mu = 0$ in (2.8), and symmetrize ΔX in the case of the XZ method.
2. Choose steplengths α, β using (2.18)–(2.19), and define

$$(7.1) \quad \sigma = \left(\frac{(X + \alpha \Delta X) \bullet (Z + \beta \Delta Z)}{X \bullet Z} \right)^3,$$

$$\mu = \sigma \frac{X \bullet Z}{n}.$$

3. Redetermine ΔX , Δy , ΔZ from (2.10), using

$$R_c = \left\{ \begin{array}{ll} \mu I - (XZ + \Delta X \Delta Z) & \text{XZ method} \\ \mu I - \frac{1}{2}(XZ + ZX + \Delta X \Delta Z + \Delta Z \Delta X) & \text{XZ + ZX method} \end{array} \right\},$$

symmetrize ΔX in the case of the XZ method, and update the iterates by

$$\begin{aligned} X &\leftarrow X + \alpha \Delta X, \\ y &\leftarrow y + \beta \Delta y, \\ Z &\leftarrow Z + \beta \Delta Z, \end{aligned}$$

with α, β given by (2.18) and (2.19).

See [16] for a definition of the PC version of the NT method. (Our experiments use (7.1) in the implementation of all the methods, although [16] uses the exponent 2 instead of 3 in (7.1).)

Computational results are presented in Tables 1 through 4. Tables 1, 2, and 3 report results for randomly generated problems, with $m = n$. The matrices $A_k, k = 1, \dots, m$ were symmetric with entries uniformly distributed in the interval $[-1, 1]$. The vector b and the matrix C were chosen to ensure that Assumption 1 was satisfied. More precisely, random positive definite symmetric matrices $\tilde{X}$ and $\tilde{Z}$ and a random vector $\tilde{y}$ were generated, and b was defined by $b_k = A_k \bullet \tilde{X}$, $k = 1, \dots, m$, while C was set to $\tilde{Z} + \sum_{k=1}^m \tilde{y}_k A_k$. All methods were initialized with the infeasible starting point $(X_0, y_0, Z_0) = (I, 0, I)$. Table 1 shows results for the XZ+ZX, XZ, and NT basic iteration, using $\sigma = 0.25$ in (2.17), with various choices for the steplength parameter τ in (2.18), (2.19). We also implemented the X^{-1} method but found it required many

TABLE 1

(a) Number of iterations to reduce gap by 10^{12} averaged over 100 randomly generated problems. Basic iteration with $\sigma = 0.25$; starting infeasible; $n = 20$, $m = 20$; S: short step failure (not included in average); E: exceeded limit failure (not included in average).

Method	$\tau = 0.9$	$\tau = 0.99$	$\tau = 0.999$
$XZ + ZX$	21.6	21.2	21.2
XZ	21.8	22.1	23.7 (S:11%, E:2%)
NT	21.6	22.0	29.8 (E:18%)

(b) Log norm infeasibility averaged over same data.

Method	$\tau = 0.9$	$\tau = 0.99$	$\tau = 0.999$
$XZ + ZX$	-12.6	-12.6	-12.6
XZ	-11.0	-10.9	-10.9
NT	-10.8	-10.7	-10.5

TABLE 2

(a) Number of iterations to reduce gap by 10^{12} averaged over 100 randomly generated problems. Mehrotra predictor-corrector rule; starting infeasible; $n = 20$, $m = 20$; S: short step failure (not included in average); E: exceeded limit failure (not included in average).

Method	$\tau = 0.9$	$\tau = 0.99$	$\tau = 0.999$
$XZ + ZX$	14.0	9.4	8.5
XZ	15.3	14.2	15.7 (S:63%, E:5%)
NT	14.5	22.8	(E:100%)

(b) Log norm infeasibility averaged over same data.

Method	$\tau = 0.9$	$\tau = 0.99$	$\tau = 0.999$
$XZ + ZX$	-10.7	-12.0	-12.2
XZ	-8.7	-8.8	-9.4
NT	-9.1	-8.3	

TABLE 3

(a) Number of iterations to reduce gap by 10^{12} averaged over 100 randomly generated problems. Mehrotra predictor-corrector rule; starting infeasible; S: short step failure (not included in average).

Method	$n = m = 20$	$n = m = 40$	$n = m = 80$
$XZ + ZX$, $\tau = 0.99$	9.4	9.9	10.0
$XZ + ZX$, $\tau = 0.999$	8.6	9.2 (S:3%)	9.5 (S:6%)

(b) Log norm infeasibility averaged over same data.

Method	$n = m = 20$	$n = m = 40$	$n = m = 80$
$XZ + ZX$, $\tau = 0.99$	-12.1	-11.2	-10.4
$XZ + ZX$, $\tau = 0.999$	-12.3	-11.4	-10.5

more iterations than the others with the same parameter choices. Table 2 shows results for the PC variants. Part (a) of both tables shows the number of iterations required to reduce the quantity $X \bullet Z$ by a factor of 10^{12} , averaged over 100 problems. Part (b) shows the final value of

$$\log_{10} (\|r_p\| + \|R_d\|),$$

averaged over the same data. A run was terminated reporting success when $X \bullet Z$ was reduced by the desired factor of 10^{12} and reporting failure if (i) the primal or dual steplength (α or β) dropped below 10^{-4} (indicated by the notation S in part (a) of the table), (ii) the number of iterations exceeded the maximum value 50 (indicated

by E in the table), or (iii) a Cholesky factorization failed (caused by rounding errors, impossible in exact arithmetic, and indicated by R in the table). Failures are *not* included in the average statistics. All experiments were conducted in Matlab, using IEEE double-precision arithmetic.

Let us first consider the results shown in Table 1 for the Basic Iteration without the PC rule. For $\tau = 0.9$, all three methods show essentially the same number of iterations. The XZ+ZX method achieves the highest accuracy (in terms of feasibility). More aggressive choices of the step-length parameter have little effect on the XZ+ZX method but cause difficulties for the XZ and NT methods. Choosing $\tau = 0.999$ causes the XZ and NT methods to fail in many cases. In the case of the XZ method, this was usually because the primal or dual steplength dropped below 10^{-4} , but for the NT method, failure generally occurred because the desired reduction in the duality gap was not achieved in 50 iterations.

Table 2 shows the same experiment using the PC rule. With $\tau = 0.9$, the PC rule greatly reduces the number of iterations, though with some loss of feasibility for the XZ and NT methods. More aggressive choices of τ gave a significantly reduced number of iterations (without loss of feasibility) for the XZ+ZX method, but led to many failures for the XZ and NT algorithms.

In Table 3, we show results for the XZ+ZX method when the problem size n is varied, using the PC rule and two choices of τ . We see an iteration count which is essentially constant as n increases, with occasional failures (with steps too short) for $\tau = 0.999$. In these cases, we found that success could generally be achieved by restarting with X_0 and Z_0 set to a larger multiple of the identity (alternatively, reducing τ). Note some loss of feasibility (due to rounding errors) for larger n . Primal feasibility can be regained by projecting onto the set $\{x : \mathbf{A}x = b\}$, but this generally fails to give a more accurate solution, as the duality gap usually increases.

For some classes of problem, the XZ and NT methods can be implemented very efficiently. This is the case, for example, for SDPs with only diagonal constraints on X (equivalently, off-diagonal entries in Z fixed). For such an SDP, we have $m = n$ and $A_k = e_k e_k^T$, $k = 1, \dots, m$, where e_k is the k th column of the identity matrix. Consequently, for the XZ method we have $\mathbf{M}_{ij} = e_i^T X e_j e_j^T Z^{-1} e_i$, i.e. $\mathbf{M}$ is the Hadamard product of X and Z^{-1} [6], reducing the cost of forming $\mathbf{M}$ from $O(n^4)$ (the general case when $m = n$) to $O(n^3)$. It is not known how to implement the XZ+ZX method efficiently in this case. A similar observation applies to the SDP that computes the Lovász θ function for a graph [5], as long as the number of edges is not too large. In this case n is the number of vertices in the graph and $m - 1$ is the number of edges, with $b = e_1$, $-C$ the matrix of all ones, $A_1 = I$, and, for $k = 2, \dots, m$, $A_k = e_i e_j^T + e_j e_i^T$, where the $(k - 1)$ th edge of the graph is from vertex i to vertex j .

Table 4 shows results comparing the XZ+ZX, XZ, and NT methods on the θ function for randomly generated graphs, with edge density 50%, using the general-purpose implementations. We set $n = 20$, so the expected value of m is $\frac{1}{4}n(n-1)+1 = 96$. For these runs, we used the initial feasible point $(X_0, y_0, Z_0) = ((1/n)I, -2ne_1, 2nI + C)$. Using an infeasible initial point did not significantly change the results. The XZ and NT methods often had difficulty reducing the duality gap by the desired factor, even with $\tau = 0.9$, because rounding errors caused a Cholesky factorization to fail. This was usually the Cholesky factorization of $\mathbf{M} = \tilde{\mathbf{A}}\tilde{\mathbf{A}}^T$ (see (2.15) and (5.3)), which is positive definite in exact arithmetic but may be numerically indefinite. Since the Schur complement for the XZ+ZX method is nonsymmetric, it is factored using

TABLE 4

(a) *Lovasz θ function: Number of iterations to reduce gap by 10^{12} averaged over 100 randomly generated problems. Mehrotra predictor-corrector rule; starting infeasible; $n = 20$, edge density 0.5 ($m \approx 96$); S: short step failure (not included in average); E: exceeded limit failure (not included in average); R: rounding failure (not included in average).*

Method	$\tau = 0.9$	$\tau = 0.99$	$\tau = 0.999$
$XZ + ZX$	15.2	11.0 (S:1%)	10.4 (S:1%)
XZ	15.5 (R:22%)	17.0 (E:15%, R:22%)	15.0 (S:68%, E:28%, R:3%)
NT	15.6 (R:25%)	21.3 (E:15%, R:22%)	(S:3%, E:97%)

(b) *Log norm infeasibility averaged over same data.*

Method	$\tau = 0.9$	$\tau = 0.99$	$\tau = 0.999$
$XZ + ZX$	-13.8	-13.7	-13.6
XZ	-12.6	-11.4	-12.4
NT	-12.5	-10.9	

an LU factorization, which fails only if the matrix is numerically singular, i.e., the factorization generates a zero pivot.

We also implemented the Q method and observed that it has essentially the same rapid local convergence and high accuracy properties as the $XZ+ZX$ method, although when initialized far from the solution it generally fails to converge.

We conclude that the $XZ+ZX$ PC method is the most efficient in terms of number of iterations, most accurate in terms of feasibility, and most robust with respect to its ability to step close to the boundary.

Appendix: Symmetric Kronecker products. Consider the linear operator on $\mathbb{R}^{n \times n}$ defined by the map

$$(7.2) \quad K \mapsto NKM^T,$$

where $M, N \in \mathbb{R}^{n \times n}$. It is standard to represent this linear operator by the Kronecker product

$$M \otimes N = \begin{bmatrix} M_{11}N & \cdots & M_{1n}N \\ \vdots & & \vdots \\ M_{n1}N & \cdots & M_{nn}N \end{bmatrix},$$

where nvec maps $\mathbb{R}^{n \times n}$ to $\mathbb{R}^{n^2}$, stacking the columns of a matrix in a vector, since then

$$(7.3) \quad (M \otimes N) \text{nvec}(K) = \text{nvec}(NKM^T).$$

Other Kronecker product identities include

$$(7.4) \quad (M \otimes N)^{-1} = M^{-1} \otimes N^{-1} \quad \text{and} \quad (M \otimes N)(K \otimes L) = MK \otimes NL.$$

Now consider the linear operator on $\mathcal{S}^n$ defined by the map

$$(7.5) \quad K \mapsto \frac{1}{2}(NKM^T + MKN^T),$$

where $M, N \in \mathbb{R}^{n \times n}$. To represent this map as a matrix, define $M \circledast N$ by the identity

$$(7.6) \quad (M \circledast N) \text{svec}(K) = \text{svec}\left(\frac{1}{2}(NKM^T + MKN^T)\right),$$

where svec maps $\mathcal{S}^n$ to $\mathbb{R}^{n^2}$ by

$$(7.7) \quad \text{svec}(K) = \left[K_{11}, \sqrt{2}K_{12}, \dots, \sqrt{2}K_{1n}, K_{22}, \dots, \sqrt{2}K_{2n}, \dots, K_{nn} \right]^T.$$

Note that

$$K \bullet L = \text{svec}(K)^T \text{svec}(L).$$

Of course, the ordering used in (7.7) is arbitrary: the important point is that each element of $\text{svec}(K)$ is associated with an index pair (i, j) , with $i \leq j$. The ordering chosen for svec dictates a corresponding ordering for $\otimes$.

We call the matrix $M \otimes N$ a *symmetric Kronecker product*. Note the identity

$$(7.8) \quad M \otimes N = N \otimes M.$$

Furthermore $(M \otimes M)^{-1} = M^{-1} \otimes M^{-1}$, but $(M \otimes N)^{-1} \neq M^{-1} \otimes N^{-1}$, in general.

We need the following lemmas whose proofs are straightforward.

LEMMA 7.1. *Let $V \in \mathbb{R}^{n \times n}$ and let v_i , $1 \leq i \leq n$ denote the columns of V . The (i, j) th column of $V \otimes V$, $1 \leq i \leq j \leq n$ is the vector*

$$\begin{cases} \text{svec}(v_i v_i^T) & \text{if } i = j, \\ \frac{1}{\sqrt{2}} \text{svec}(v_i v_j^T + v_j v_i^T) & \text{if } i < j. \end{cases}$$

LEMMA 7.2. *Let M, N be commuting symmetric matrices, and let $\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n$ denote their eigenvalues with $v_1, \dots, v_n$ a common basis of orthonormal eigenvectors. The $n(n+1)/2$ eigenvalues of $M \otimes N$ are given by*

$$\frac{1}{2}(\alpha_i \beta_j + \beta_i \alpha_j), \quad 1 \leq i \leq j \leq n,$$

with the corresponding set of orthonormal eigenvectors

$$\begin{cases} \text{svec}(v_i v_i^T) & \text{if } i = j, \\ \frac{1}{\sqrt{2}} \text{svec}(v_i v_j^T + v_j v_i^T) & \text{if } i < j. \end{cases}$$

In other words, if $V = [v_1 \cdots v_n]$, then $V \otimes V$ is an orthogonal matrix of size $n^2 \times n^2$ which diagonalizes $M \otimes N$. The standard algorithm for solving the Lyapunov equation $MXN^T + NXM^T = B$ (when M and N commute) immediately follows: the solution is VCV^T , where C is found by computing $V^T B V$ and dividing its entries by the quantities $(\alpha_i \beta_j + \beta_i \alpha_j)$ componentwise.

Acknowledgments. The third author is grateful to Richard Tapia, Bob Vanderbei, and Steve Wright for introducing him to primal-dual interior-point methods for LP. The authors also thank Mike Todd for many helpful comments, especially a suggestion concerning the Q method that makes the development given here substantially simpler than the one given in [1].

REFERENCES

- [1] F. ALIZADEH, J.-P. A. HAEBERLY, AND M. L. OVERTON, *A new primal-dual interior-point method for semidefinite programming*, in Proc. 5th SIAM Conference on Applied Linear Algebra, J.G. Lewis, ed., SIAM, Philadelphia, 1994, pp. 113–117.

- [2] F. ALIZADEH, J.-P. A. HAEBERLY, AND M. L. OVERTON, *Primal-dual interior-point methods for semidefinite programming*, manuscript presented at the Math. Programming Symposium, Ann Arbor, MI, 1994.
- [3] F. ALIZADEH, J.-P. A. HAEBERLY, AND M. L. OVERTON, *Complementarity and nondegeneracy in semidefinite programming*, Math. Programming, 77 (1997), pp. 111–128.
- [4] F. ALIZADEH, *Combinatorial Optimization with Interior Point Methods and Semidefinite Matrices*, Ph.D. thesis, University of Minnesota, Minneapolis, MN, 1991.
- [5] M. GRÖTSCHEL, L. LOVÁSZ, AND A. SCHRIJVER, *Geometric Algorithms and Combinatorial Optimization*, Springer-Verlag, New York, 1988.
- [6] C. HELMBERG, F. RENDL, R. VANDERBEI, AND H. WOLKOWICZ, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.
- [7] M. KOJIMA, S. SHINDOH, AND S. HARA, *Interior-point methods for the monotone linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.
- [8] M. KOJIMA, M. SHIDA, AND S. SHINDOH, *A predictor-corrector interior-point algorithm for the semidefinite linear complementarity problem using the Alizadeh-Haeberly-Overton search direction*, SIAM J. Optim., to appear.
- [9] R. MONTEIRO, *Polynomial convergence of primal-dual algorithms for semidefinite programming based on Monteiro and Zhang family of directions*, SIAM J. Optim., 8 (1998), pp. 797–812.
- [10] R. MONTEIRO AND P. ZANJACOMO, *A note on the existence of the Alizadeh-Haeberly-Overton direction for semidefinite programming*, Math. Programming, 78 (1997), pp. 393–396.
- [11] Y. NESTEROV AND A. NEMIROVSKII, *Interior Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, 1994.
- [12] Y. NESTEROV AND M. TODD, *Primal-dual interior-point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.
- [13] Y. NESTEROV AND M. TODD, *Self-scaled barriers and interior-point methods for convex programming*, Math. Oper. Res., 22 (1997), pp. 1–42.
- [14] M. L. OVERTON AND R. S. WOMERSLEY, *Second derivatives for optimizing eigenvalues of symmetric matrices*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 697–718.
- [15] M. SHIDA, S. SHINDOH, AND M. KOJIMA, *Existence and uniqueness of search directions in interior-point algorithms for the SDP and the monotone SDLCP*, SIAM J. Optim., 8 (1998), pp. 387–398.
- [16] M. J. TODD, K. C. TOH, AND R. H. TÜTÜNCÜ, *On the Nesterov-Todd direction in semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 769–796.
- [17] L. VANDENBERGHE AND S. BOYD, *Primal-dual potential reduction method for problems involving matrix inequalities*, Math. Programming B, 69 (1995), pp. 205–236.
- [18] S. J. WRIGHT, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, 1997.
- [19] Y. ZHANG, *On extending some primal-dual interior-point algorithms from linear programming to semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 365–386.
- [20] Y. ZHANG, R. A. TAPIA, AND J. E. DENNIS, *On the superlinear and quadratic convergence of primal-dual interior point linear programming algorithms*, SIAM J. Optim., 2 (1992), pp. 304–324.

ON THE NESTEROV–TODD DIRECTION IN SEMIDEFINITE PROGRAMMING*

M. J. TODD[†], K. C. TOH[‡], AND R. H. TÜTÜNCÜ[§]

Abstract. We study different choices of search direction for primal-dual interior-point methods for semidefinite programming problems. One particular choice we consider comes from a specialization of a class of algorithms developed by Nesterov and Todd for certain convex programming problems. We discuss how the search directions for the Nesterov–Todd (NT) method can be computed efficiently and demonstrate how they can be viewed as Newton directions. This last observation also leads to convenient computation of accelerated steps, using the Mehrotra predictor-corrector approach, in the NT framework. We also provide an analytical and numerical comparison of several methods using different search directions, and suggest that the method using the NT direction is more robust than alternative methods.

Key words. semidefinite programming, Newton direction, predictor-corrector interior-point method

AMS subject classification. 90C05

PII. S105262349630060X

1. Introduction. In this paper, we study interior-point methods for semidefinite programming (SDP) problems based on the search direction described by Nesterov and Todd [17, 18]. For simplicity, we call this direction the Nesterov–Todd (NT) direction. We consider the SDP problem given in the standard form

$$(1) \quad \begin{aligned} \text{(SDP)} \quad \min_X \quad & C \bullet X, \\ & A_i \bullet X = b_i, \quad i = 1, \dots, m, \\ & X \succeq 0, \end{aligned}$$

where each $A_i \in \mathbb{S}^{n \times n}$, $b \in \mathbb{R}^m$, $C \in \mathbb{S}^{n \times n}$ are given, and $X \in \mathbb{S}^{n \times n}$. Here $\mathbb{S}^{n \times n}$ denotes the space of $n \times n$ symmetric matrices, and $X \succeq 0$ indicates that X is positive semidefinite. We assume that the set $\{A_i\}$ is linearly independent. The dual problem associated with (SDP) is

$$(2) \quad \begin{aligned} \text{(SDD)} \quad \max_{y, S} \quad & b^T y, \\ & \sum_{i=1}^m y_i A_i + S = C, \\ & S \succeq 0, \end{aligned}$$

*Received by the editors March 14, 1996; accepted for publication (in revised form) May 7, 1997; published electronically July 20, 1998.

<http://www.siam.org/journals/siopt/8-3/30060.html>

[†]School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY 14853 (miketodd@cs.cornell.edu). The research of this author was supported in part by NSF grant DMS-9505155 and ONR grant N00014-96-1-0050.

[‡]Department of Mathematics, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119620 (matttohc@math.nus.sg). The research of this author was supported by a graduate scholarship from the National University of Singapore.

[§]Department of Mathematical Sciences, Carnegie Mellon University, Pittsburgh, PA 15213 (reha@qwe4.math.cmu.edu). The research of this author was supported in part by NSF grant DMS-9505155 and ONR grant N00014-96-1-0050 while the author was a graduate student at Cornell University.

where $y \in \mathbb{R}^m$ and $S \in S\mathbb{R}^{n \times n}$.

SDP problems have a wide range of applications in both continuous and combinatorial optimization (we refer the reader to [22] for an extensive list of applications). Interior-point methods for their solution were pioneered by Alizadeh [1] and Nesterov and Nemirovskii [15, 16] independently. These methods were primal; more efficient algorithms use a primal-dual approach. For a comprehensive list of publications concerned with both algorithms and applications and related software, see the semidefinite programming home pages maintained by Alizadeh [2] and Helmberg [6].

Our goal is to develop efficient algorithms (in terms of both number of iterations and number of arithmetic operations per iteration) for a wide range of classes of SDP problems. We will see that methods based on the NT direction compare favorably with other interior-point methods. One issue that is not addressed here is the exploitation of problem structure for large-scale problems. All the methods considered here use direct methods that do not preserve sparsity to solve the linear systems that arise in each iteration, while indirect methods may be preferable generally. However, if the $\{A_i\}$ and C have a common block-diagonal structure, then so do all feasible X and S , and all the algorithms here preserve and exploit this structure. In particular, the linear systems that arise can all be solved blockwise.

As in linear programming (LP), most interior-point algorithms for SDP problems can be viewed as damped and centered Newton methods. At each iteration, the search direction is computed by applying Newton's method to a system of nonlinear equations defining the so-called central path. However, unlike the case for LP, even for primal-dual methods, there are several ways to write these nonlinear equations. Indeed, for some choices, the direction that results from the Newton equation does not give symmetric matrices as the search directions for X and S , and hence cannot be implemented without further modification. One way to proceed is to symmetrize the complementarity equation in the system defining the central path with respect to an invertible matrix P , as was done by Monteiro [14] and Y. Zhang [23]. The resulting Newton system gives rise to different search directions for different matrices P . We give a simple proof that the symmetrized Newton system has a unique solution under certain conditions related to those appearing in the work of Shida, Shindoh, and Kojima [19]. Based on this uniqueness result, we show that the NT direction can be expressed as the solution of a symmetrized Newton system. (We note that recently, independently of our work, Sturm and S. Zhang [20] described how to generate the NT direction as a Newton search direction for primal-dual path-following algorithms for SDP problems using a symmetric primal-dual transformation, similar to section 6 in [17].)

In LP, the most computationally successful interior-point methods have been primal-dual methods using Mehrotra's [12] predictor-corrector steps. These methods obtain higher-order directions from the nonlinear equations formed at each iteration. Because of our interpretation of the NT direction as a Newton direction, we can follow this approach also. In contrast, from the equation defining the NT direction in [17, 18], it is not clear how we can incorporate higher-order (quadratic) terms into the NT direction.

One of the main contributions of this paper is the implementation of such a predictor-corrector algorithm based on the NT direction, where the direction is computed from the solution of a linear least-squares problem via a QR factorization. For simplicity, we will refer to this method as Algorithm NT-PC-QR. We discuss issues related to efficient implementation, which are briefly described next. First, in solving

for the NT direction of an SDP problem, we need to compute a certain scaling matrix W . A direct computation would involve two different matrix square roots. Here we show that W can be computed with two Cholesky factorizations and one singular value decomposition (SVD). Second, we show how to use these to compute the search direction efficiently. Finally, computation of the quadratic correction requires the solution of a Lyapunov equation, but this can be solved explicitly and simply by reusing the Cholesky factors and SVD already generated.

For the purpose of comparison, we also implemented three other methods using exactly the same algorithmic framework as Algorithm NT-PC-QR but computing the search direction from a Schur complement equation via a Cholesky or an LU factorization. (This approach is like solving the normal equations of the linear least-squares problem mentioned above.) The first is based on the direction of Alizadeh, Haeberly, and Overton [3]; the second on one of the H..K..M directions; and the last on the NT direction. The second direction has a convoluted history: it apparently first appeared in a preprint of Helmberg et al. [7] in late 1993, was rediscovered independently as one member of a family of directions introduced in early 1994 by Kojima, Shindoh, and Hara [10], and was most recently derived (as well as a “dual” direction, also a member of the KSH family) in a different way by Monteiro [14] in 1995. This last derivation fits most conveniently with our discussions, but we have chosen a name that suggests all those who obtained it. These three methods are referred to as Algorithm AHO-PC-Sch, H..K..M-PC-Sch and NT-PC-Sch, respectively.

After the release of the first version of this paper, we observed that with careful implementations, the three algorithms that are based on the Schur complement approach can perform much better than we initially expected, obtaining duality gaps below 10^{-10} in the problems we tested.

Our numerical results indicate that the methods (both the QR and Schur complement approaches) based on the NT direction are fast and robust. For most problems, the Schur complement approach is as stable as the QR approach. However, it is known that the latter approach can be much better conditioned than the former. Conceivably, there may be problems where the QR approach can perform much better than the Schur complement approach because of this.

A brief summary of how the NT methods compare to the methods based on the AHO and H..K..M directions is as follows. The NT methods are almost as efficient as Algorithm AHO-PC-Sch in terms of the number of iterations required to attain a given accuracy, typically taking only two or three more iterations. Algorithm H..K..M-PC-Sch usually takes almost the same number of iterations as the NT methods, but occasionally it stagnates, taking very small steps, and then it requires several more steps. On the other hand, each iteration of the NT methods is about 50% cheaper than those of Algorithm AHO-PC-Sch and only slightly more expensive than those of Algorithm H..K..M-PC-Sch. Overall, the NT methods seem to be more robust than the other two algorithms in the sense that problems of stagnation appear less likely to occur. Finally, we must mention that Algorithm AHO-PC-Sch seems to be the best of the tested methods in terms of reducing the duality gap, which is typically 10 to 100 times smaller than that obtained from methods based on the H..K..M and NT directions.

This paper is organized as follows. In section 2, we introduce the NT direction for SDP problems and study some of its symmetry and invariance properties. In section 3, we describe the symmetrized Newton system and prove that it has a unique solution under a certain positive semidefiniteness condition. Then we apply this uniqueness

result to show that the NT direction can be viewed as the search direction of a certain symmetrized Newton system. We address implementation issues in section 4: we give an efficient way to compute the scaling matrix W , show how the Schur complement equation can be expressed as a linear least-squares problem, and derive the Mehrotra-type corrector [12] for the NT direction. Results of numerical experiments are given in section 5. Finally, we give an appendix containing some frequently used properties of the standard and symmetrized Kronecker products in the context of SDP problems.

Throughout this paper, $\mathbb{R}^n$ denotes the n -dimensional Euclidean space and $\|\cdot\|$ denotes the Euclidean norm. The standard inner product between two real $n \times n$ matrices G and K is the number $G \bullet K := \text{Tr } G^T K$, the trace of the matrix $G^T K$. A matrix $G \in \mathbb{R}^{n \times n}$ (not necessarily symmetric) is said to be positive definite (resp., positive semidefinite) if $v^T G v > 0$, ($v^T G v \geq 0$, resp.) for all $v \neq 0$. However, all positive (semi)definite matrices considered here are symmetric except in the statement and proof of Theorem 3.1, so we shall assume that positive (semi)definiteness implies symmetry unless explicitly stated otherwise. We write $G \succ 0$ ($G \succeq 0$) if G is positive definite (positive semidefinite). For $G \succeq 0$, $G^{1/2}$ denotes the unique symmetric positive semidefinite square root of G . Finally, $\mathcal{N}(G)$ and $\mathcal{R}(G)$ denote the null space and range space of the matrix G , respectively.

2. The NT direction and some symmetry and invariance issues. In two papers [17, 18], Nesterov and Todd study interior-point methods for convex programming problems expressed in conic form. They develop efficient primal-dual algorithms for a class of problems for which the cone and its associated barrier are self-scaled. This class includes semidefinite programming. All the primal-dual methods in [17, 18] use a direction that, for the case of SDP problems, is (up to a scalar multiple) the solution for some σ to the system of equations

$$\begin{aligned} A_i \bullet \Delta X &= 0, \quad i = 1, \dots, m, \\ (3) \quad \sum_{i=1}^m \Delta y_i A_i + \Delta S &= 0, \\ W^{-1} \Delta X W^{-1} + \Delta S &= \sigma \mu X^{-1} - S, \end{aligned}$$

where W is the scaling matrix defined below and $\mu := X \bullet S/n$. The unique solution to (3) is an affine-scaling direction for $\sigma = 0$, and a centering direction for $\sigma = 1$ (see [18]).

We start by describing how (3) arises. In fact, it is a special case for semidefinite programming of a system of equations defined for convex programming problems in conical form, where the cone, say K , has a *self-scaled* barrier function F . In our case, the cone K is the set of positive semidefinite matrices of order n , and the barrier function is defined by

$$(4) \quad F(X) := -\ln \det(X).$$

We will not give the definition of self-scaled barriers here (for details, see [17]), but note that a key consequence is that for every X in the interior of the primal cone K and every S in the interior of the corresponding dual cone K^* there is a unique *scaling* point W in the interior of K with

$$(5) \quad F''(W)X = S.$$

The theory also implies that, with $T := -F'(W)$, the dual function F_* , which is a self-scaled barrier for the dual cone K^* , satisfies $F''_*(T) = [F''(W)]^{-1}$ and $F''_*(T)S = X$, and we therefore have a dual scaling point. Further, we have $F''(W)F'_*(S) = F'(X)$.

In the case of semidefinite programming, $F_*(S) = -\ln \det(S) - n$, $F'(X) = -X^{-1}$, and similarly $F'_*(S) = -S^{-1}$, and $F''(W)$ is the operator taking X to $W^{-1}XW^{-1}$, and similarly for $F''_*(T)$. Equation (5) then becomes

$$(6) \quad W^{-1}XW^{-1} = S,$$

or equivalently $WSW = X$, from which we easily see that

$$(7) \quad W = X^{1/2}(X^{1/2}SX^{1/2})^{-1/2}X^{1/2} = S^{-1/2}(S^{1/2}XS^{1/2})^{1/2}S^{-1/2}.$$

The scaling matrix W is related to various notions of geometric mean for symmetric positive definite matrices. Indeed, (6) shows that W is the metric geometric mean of X and S^{-1} (and W^{-1} that of S and X^{-1}) and

$$(8) \quad W^{-1/2}XW^{-1/2} = W^{1/2}SW^{1/2}$$

is the spectral geometric mean of X and S [4, 8].

Thus system (3) amounts to feasibility equations for the primal and dual directions, together with the equation

$$(9) \quad F''(W)\Delta X + \Delta S = -S - \sigma\mu F'(X).$$

From the results above, this is equivalent to

$$(10) \quad \Delta X + F''_*(T)\Delta S = -X - \sigma\mu F'_*(S).$$

One nice feature of the NT approach is that it provides general methods applying to any convex programming problem in conic form, as long as the cone is self-scaled; such cones include the nonnegative orthant and the second-order cone, in addition to the cone of positive semidefinite matrices. Thus the convergence proofs and complexity estimates apply to this general class of problems. In contrast, the other methods considered here are specific to SDP problems, and require new convergence proofs. (However, the algorithms we implement here employ second-order directions and long steps for which the NT convergence proofs do not apply.)

In the literature on interior-point methods for semidefinite programming, one key point is the need for well-defined search directions, with ΔX and ΔS symmetric. This will be discussed further in the next section. We point out that the general development of [17, 18] leads to nonsingular linear systems defined on the appropriate linear spaces, namely the set of symmetric matrices of order n for ΔX and ΔS in our case, so that (3) has a unique solution with the directions lying in the correct spaces.

There is also an issue of primal-dual symmetry. Since (SDP) can be converted into (SDD) and vice versa, there is a natural symmetry between primal and dual. We would like our algorithms also to be symmetric between primal and dual. Equations (9) and (10) above show that (apart from the lack of symmetry in the feasibility equations, which follows from the way in which the constraints are written) the NT direction does possess this primal-dual symmetry. Some of the methods we discuss below do not share this property.

There is a third property which we believe is desirable for interior-point methods: scale invariance. For linear programming, it is well known that if the primal variables are measured in different units, then the search direction scales appropriately. That is, if x is replaced by $D^{-1}x$ where D is diagonal and positive definite, and correspondingly A becomes AD , c becomes Dc , and s becomes Ds , then the search direction

for any algorithm at the scaled trial point $(D^{-1}x, y, Ds)$ in the scaled problem is $(D^{-1}\Delta x, \Delta y, D\Delta s)$, where $(\Delta x, \Delta y, \Delta s)$ is the search direction at the unscaled point (x, y, s) in the original problem.

We will say that a search direction for SDP enjoys *scale invariance* if a similar property holds. Now we let U be any invertible $n \times n$ matrix and consider the “scaling” where X is replaced by $\bar{X} := U^{-1}XU^{-T}$ and correspondingly each A_i by $\bar{A}_i := U^T A_i U$, C by $\bar{C} := U^T C U$, and S by $\bar{S} := U^T S U$. This defines a new, scaled problem: if (X, y, S) is feasible in the original problem and its dual, $(\bar{X}, y, \bar{S})$ is feasible in the scaled problem with the same objective function values. If the search direction at the unscaled point (X, y, S) in the original problem is $(\Delta X, \Delta y, \Delta S)$, we require the search direction at the scaled point $(\bar{X}, y, \bar{S})$ in the scaled problem to be $(\bar{\Delta X}, \Delta y, \bar{\Delta S}) := (U^{-1}\Delta XU^{-T}, \Delta y, U^T \Delta S U)$.

It is not hard to show that the NT direction is scale invariant. In fact, this follows from the coordinate-free representation of the directions, but a direct proof is easy and instructive. From (6), it is easy to check that the scaling point corresponding to $U^{-1}XU^{-T}$ and $U^T S U$ is $U^{-1}WU^{-T}$ where W is the scaling point corresponding to X and S . Now it is merely a matter of substituting the scaled values of $\{A_i\}$, ΔX , ΔS , and W into (3) to confirm that the solutions scale appropriately.

Note that if we choose U above to be $W^{1/2}$, then both X and S are scaled to the same point by (8). This is parallel to the situation in LP. In this scaled space, the search directions for X and S are just Euclidean projections of the scaled right-hand side onto the appropriate orthogonal subspaces; see section 6 of [17] for a general view of this. This scaling is the starting point for the development in Sturm and S. Zhang [20].

3. The central path and the Newton equation.

3.1. The Newton equation and symmetric search directions. For the SDP problem, the central path is defined as the set of solutions to the equations

$$(11) \quad \begin{aligned} A_i \bullet X &= b_i \quad \text{for } i = 1, \dots, m, \\ \sum_{i=1}^m y_i A_i + S &= C, \\ XS &= \nu I \end{aligned}$$

for all $\nu > 0$ (together with the requirement that $X \succ 0$, $S \succ 0$). Given an iterate (X, y, S) (not necessarily feasible) for the primal and dual SDP problems, many interior-point algorithms generate the next iterate by taking a single Newton step applied to the nonlinear system of equations (11) or a related system. Much of the recent debate on interior-point methods for SDP problems has focused on the issue of the linearization of the third equation in this system. Suppose we try to approximate the point on the central path corresponding to a value $\nu = \sigma\mu$, where $\mu := X \bullet S/n$ and $\sigma \in [0, 1]$. If the solution is $(X + \Delta X, y + \Delta y, S + \Delta S)$, then we have

$$(12a) \quad A_i \bullet \Delta X = b_i - A_i \bullet X \quad \text{for } i = 1, \dots, m,$$

$$(12b) \quad \sum_{i=1}^m \Delta y_i A_i + \Delta S = C - S - \sum_{i=1}^m y_i A_i,$$

$$(12c) \quad \Delta XS + X \Delta S = \sigma\mu I - (XS + \Delta X \Delta S).$$

The linearization of (12c) gives

$$(12d) \quad \Delta XS + X \Delta S = \sigma\mu I - XS.$$

A key requirement for interior-point methods for semidefinite programming is to generate symmetric ΔX and ΔS matrices. By virtue of (12b), any ΔS that satisfies the above system has to be symmetric. However, this is not true in general for ΔX . Even when the system (12a), (12b), and (12d) is feasible, there may not be any solutions with symmetric ΔX (see [10] for an example). Different recipes have been suggested to symmetrize (12c) and (12d). We follow Y. Zhang's approach [23], that unifies these symmetrizations by viewing them as parametrized transformations of (12c) and (12d).

In [23], Zhang defines the linear transformation

$$(12e) \quad H_P(M) := \frac{1}{2} [PMP^{-1} + P^{-T}M^TP^T]$$

for a given invertible matrix P and observes that if P is invertible and M is similar to a (symmetric) positive definite matrix, then

$$H_P(M) = \nu I \Leftrightarrow M = \nu I.$$

This observation indicates that the last equation in (11) can be replaced by

$$(12f) \quad H_P(XS) = \nu I$$

without affecting the definition of the central path, since XS is similar to the positive definite matrix $S^{1/2}XS^{1/2}$. Using this characterization of the central path, (12c) and (12d) are replaced by

$$(12c') \quad H_P(\Delta X S + X \Delta S) = \sigma \mu I - H_P(XS + \Delta X \Delta S)$$

and

$$(12d') \quad H_P(\Delta X S + X \Delta S) = \sigma \mu I - H_P(XS).$$

The resulting system (12a), (12b), and (12d') can be viewed as defined by a linear transformation from $\mathbb{R}^m \times S\mathbb{R}^{n \times n} \times S\mathbb{R}^{n \times n}$ to itself, and thus under suitable conditions it will produce a unique solution $(\Delta y, \Delta X, \Delta S)$ with symmetric ΔX and ΔS . Therefore, we will restrict ΔX to be symmetric in the rest of our discussion. Note that the equation (12d') can then be written in full detail as

$$(13) \quad \begin{aligned} & P(\Delta X S + X \Delta S)P^{-1} + P^{-T}(S \Delta X + \Delta S X)P^T \\ & = 2\sigma \mu I - PXS P^{-1} - P^{-T}SXP^T. \end{aligned}$$

Alizadeh, Haeberly, and Overton's linearization [3] corresponds to $P = I$, and Monteiro's two linearizations [14] to $P = X^{-1/2}$ and $P = S^{1/2}$. In fact, the second of Monteiro's linearizations gives the same direction as that introduced by Helmberg et al. [7], and both are included in a class developed by Kojima, Shindoh, and Hara [10]; see also Lin and Saigal [11]. However, the motivation of [7, 10] is very different, and to view all these directions in a unified way we follow Monteiro's approach. We call these the AHO direction and the H.K.M directions. In section 3.4, we shall see that the NT direction also results from such a linearization for a suitable P .

3.2. Symmetrized Kronecker product notation. To compute the Newton step $(\Delta X, \Delta y, \Delta S)$, it is easier to express the linear systems of equations (12a), (12b), and (12d') in the standard matrix-vector form by using symmetrized Kronecker products. First we define an operator taking symmetric matrices into vectors: if U is an $n \times n$ symmetric matrix, $\text{svec}(U)$ is defined by

$$(14) \quad \text{svec}(U) := (u_{11}, \sqrt{2}u_{21}, \dots, \sqrt{2}u_{n1}, u_{22}, \sqrt{2}u_{32}, \dots, \sqrt{2}u_{n2}, \dots, u_{nn})^T.$$

The factor $\sqrt{2}$ is introduced so that svec is an isometry between $S\mathbb{R}^{n \times n}$ and $\mathbb{R}^{n(n+1)/2}$ with their respective standard inner products. We denote the inverse map of svec by smat .

The symmetrized Kronecker product of any two $n \times n$ matrices G and K (not necessarily symmetric) is a square matrix of order $n(n+1)/2$; its action on a vector $u := \text{svec}(U)$, where $U \in S\mathbb{R}^{n \times n}$, is given by

$$(15) \quad (G \otimes_s K) \text{svec}(U) := \frac{1}{2} \text{svec}(KUG^T + GUK^T).$$

The operator svec and the symmetrized Kronecker product were first used by Alizadeh, Haerberly, and Overton [3], but with the latter restricted to the case where G and K were both symmetric. We refer the reader to the appendix for some properties of the symmetrized Kronecker product, as well as corresponding properties for, and relations to, the standard Kronecker product.

Using (14) and (15), we can now write the system (12a), (12b), and (12d') in a 3×3 block equation

$$(16) \quad \begin{pmatrix} 0 & \mathcal{A} & 0 \\ \mathcal{A}^T & 0 & \mathcal{I} \\ 0 & E & F \end{pmatrix} \begin{pmatrix} \Delta y \\ \text{svec}(\Delta X) \\ \text{svec}(\Delta S) \end{pmatrix} = \begin{pmatrix} r_p \\ \text{svec}(R_d) \\ \text{svec}(R_c) \end{pmatrix},$$

where $\mathcal{I}$ is the identity matrix of order $n(n+1)/2$,

$$(17) \quad E := P \otimes_s P^{-T} S, \quad F := PX \otimes_s P^{-T},$$

$$(18) \quad r_p := b - \mathcal{A} \text{svec}(X), \quad R_d := C - S - \sum_{i=1}^m y_i A_i, \quad R_c := \sigma \mu I - H_P(XS).$$

Here, $\mathcal{A}$ is a matrix of dimension $m \times n(n+1)/2$ defined by

$$(19) \quad \mathcal{A}^T := [\text{svec}(A_1) \cdots \text{svec}(A_m)].$$

The assumption that the $\{A_i\}$ are linearly independent implies that $\mathcal{A}$ has full row rank.

3.3. Uniqueness of symmetric search directions. For the rest of our presentation it is essential to understand the conditions under which the system (16) has a unique solution with symmetric ΔX and ΔS . In their paper [19], Shida, Shindoh, and Kojima established the existence and uniqueness of the solutions of (16) for some choices of the matrix P , as well as some other systems that may not be expressible in the form (16). They show that when $P = X^{-1/2}$, the system (16) always has a unique solution as long as $X \succ 0$ and $S \succ 0$, but that this is not true for $P = I$: in this case the additional condition that $XS + SX \succeq 0$ is derived as a sufficient condition for (16) to have a unique solution. This last condition turns out to be sufficient but not necessary. Before generalizing their result to arbitrary invertible matrices P , we observe that the matrices E and F are nonsingular when X and S are positive definite. Indeed,

$$(20) \quad E = P \otimes_s P^{-T} S = (I \otimes_s P^{-T} S P^{-1}) (P \otimes_s P)$$

and

$$(21) \quad F = PX \otimes_s P^{-T} = (PXP^T \otimes_s I) (P^{-T} \otimes_s P^{-T}),$$

where the equalities follow from the properties of the symmetrized Kronecker product listed in the appendix. Now, E and F are nonsingular since both factors in (20) and (21) are nonsingular (see the appendix).

THEOREM 3.1. *Suppose X and S are positive definite. Then the system of equations (16) has a unique solution $(\Delta y, \Delta X, \Delta S) \in \mathbb{R}^m \times S\mathbb{R}^{n \times n} \times S\mathbb{R}^{n \times n}$ if $E^{-1}F$ is positive definite (not necessarily symmetric). In particular, this condition holds when X , S , and $H_P(XS)$ are positive semidefinite.*

Proof. For this proof, positive definiteness does not imply symmetry; if matrices are symmetric, we will explicitly say so.

It suffices to show that the 3×3 block system (with a square matrix)

$$(22) \quad \begin{pmatrix} 0 & \mathcal{A} & 0 \\ \mathcal{A}^T & 0 & \mathcal{I} \\ 0 & E & F \end{pmatrix} \begin{pmatrix} \Delta y \\ \text{svec}(\Delta X) \\ \text{svec}(\Delta S) \end{pmatrix} = 0$$

has only the trivial solution.

Since E is invertible by (20), we can use block Gaussian elimination to reduce (22) to a Schur complement equation given by

$$(23) \quad (\mathcal{A}E^{-1}F\mathcal{A}^T)\Delta y = 0.$$

Since $E^{-1}F$ is assumed to be positive definite and $\mathcal{A}$ has full row rank, this implies that $\mathcal{A}E^{-1}F\mathcal{A}^T$ is positive definite and thus $\Delta y = 0$ is the only solution to (23).

Now the second block equation in (22) gives $\Delta S = -\text{smat}(\mathcal{A}^T \Delta y) = 0$, and the third block equation gives $\Delta X = -E^{-1}F\Delta S = 0$. This shows that the solution to (16) exists and is unique.

To show that the condition $H_P(XS)$ is positive semidefinite implies that $E^{-1}F$ is positive definite, consider an arbitrary $n(n+1)/2$ -dimensional nonzero vector g . We shall prove that $g^T E^{-1}Fg > 0$. Let $k := E^{-T}g$ and $K := \text{smat}(k)$, so that k is also nonzero. We have

$$\begin{aligned} g^T E^{-1}Fg &= k^T F E^T k = k^T (PX \otimes_s P^{-T})(P^T \otimes_s SP^{-1})k \\ &= \frac{1}{2} k^T (PXP^T \otimes_s P^{-T}SP^{-1})k + \frac{1}{2} k^T (PXSP^{-1} \otimes_s I)k \\ &> \frac{1}{2} k^T (PXSP^{-1} \otimes_s I)k \\ &= \frac{1}{4} (\text{svec}K)^T \text{svec}(PXSP^{-1}K + KP^{-T}SXP^T) \\ &= \frac{1}{4} \text{Tr} [K(PXSP^{-1} + P^{-T}SXP^T)K] \\ &= \frac{1}{2} \text{Tr} [K H_P(XS) K] \geq 0. \end{aligned}$$

The strict inequality above holds because $PXP^T \otimes_s P^{-T}SP^{-1}$ is positive definite, a consequence of the fact that PXP^T and $P^{-T}SP^{-1}$ are symmetric positive definite (see the appendix). The last inequality follows from the assumption that $H_P(XS)$ is positive semidefinite. This completes the proof. $\square$

We note that, if $P = X^{-1/2}$ or, more generally, $P = (X^{1/2}SX^{1/2})^\alpha X^{-1/2}$ for some α , then $H_P(XS) = X^{1/2}SX^{1/2}$, which is positive definite whenever X and S are positive definite. Similarly, when $P = S^{1/2}$ or $P = (S^{1/2}XS^{1/2})^\alpha S^{1/2}$, we have $H_P(XS) = S^{1/2}XS^{1/2}$, which again is positive definite whenever X and S are. When

$P = I$, the sufficient condition given in Theorem 3.1 (i.e., that $H_P(XS)$ is positive semidefinite) coincides with the condition in [19]. So, as in [19], we see that the system (16) always has a unique solution with symmetric ΔX and ΔS for Monteiro's choices of P . This also holds for the choice of Alizadeh, Haeberly, and Overton [3] of P if $XS + SX$ is positive semidefinite, but there are examples where the coefficient matrix in (16) is singular. One such example is given by

$$A = \begin{bmatrix} -1 & 2 & 0 \end{bmatrix}, \quad X = \begin{bmatrix} 1 & \sqrt{2} \\ \sqrt{2} & 3 \end{bmatrix}, \quad S = \begin{bmatrix} 1 & 0 \\ 0 & 11 \end{bmatrix}.$$

Indeed, in this case the Schur complement matrix in (23) is the 1×1 zero matrix.

The following theorem describes a class of matrices P that satisfy the sufficient condition of Theorem 3.1.

THEOREM 3.2. *Suppose X and S are positive definite matrices and P is an invertible matrix. Let E and F be as defined in (17). Then the following are equivalent:*

- (a) PXP^T and $P^{-T}SP^{-1}$ commute;
- (b) $PXSP^{-1}$ is symmetric;
- (c) FE^T is symmetric; and
- (d) $E^{-1}F$ is symmetric.

Condition (a) implies that $H_P(XS)$ is positive definite. Consequently, any of the conditions above imply that $E^{-1}F$ and FE^T are both positive definite.

Proof. The equivalence of (a) and (b) is immediate: PXP^T and $P^{-T}SP^{-1}$ commute if and only if

$$(24) \quad PXSP^{-1} = (PXP^T)(P^{-T}SP^{-1}) = (P^{-T}SP^{-1})(PXP^T) = P^{-T}SXP^T.$$

On the other hand,

$$FE^T = (PX \otimes_s P^{-T})(P^T \otimes_s SP^{-1}) = \frac{1}{2}(PXP^T \otimes_s P^{-T}SP^{-1} + PXSP^{-1} \otimes_s I).$$

The first term on the right-hand side above is symmetric, since PXP^T and $P^{-T}SP^{-1}$ are. Therefore FE^T is symmetric if and only if the second term is. But this holds if and only if $PXSP^{-1}$ is symmetric. This shows the equivalence of (b) and (c). Finally, the equivalence of (c) and (d) is obvious: $E^{-1}F = E^{-1}(FE^T)E^{-T}$ is symmetric if and only if FE^T is.

Now suppose (a) holds (thus (b) holds). Then

$$H_P(XS) = PXSP^{-1} = (PXP^T)(P^{-T}SP^{-1}),$$

which is the product of two commuting positive definite matrices. This implies that $H_P(XS)$ is positive definite. Consequently, by Theorem 3.1, $E^{-1}F$ and hence FE^T are positive definite. $\square$

The first of the equivalent conditions of Theorem 3.2 holds for the two Monteiro choices for P (one of the two matrices is the identity), and also for $P = W^{-1/2}$ (the two matrices are equal from (8)); we will see much more of this choice in the rest of the paper.

Of the two H..K..M directions and the AHO direction, only the latter is primal-dual symmetric. Turning now to scale invariance, we see that the AHO direction is *not* scale invariant; XS transforms to $U^{-1}XSU$ while SX transforms to $U^T SXU^{-T}$, and similarly for products involving directions.

TABLE 1
Properties of the AHO, H..K..M, and NT directions.

Direction	Primal-dual symmetry	Scale invariance	Directions uniquely defined
AHO ($P = I$) [3]	yes	no	no
H..K..M ($P = X^{-1/2}$ or $S^{1/2}$) [7, 10, 14]	no	yes	yes
NT ($P = W^{-1/2}$) [17, 18]	yes	yes	yes

To see that the other two directions are scale invariant, we note first that the solution to (16) corresponding to P is the same as that corresponding to QP for any orthogonal Q ; indeed, the direction depends only on $P^T P$, as is easily seen by pre- and postmultiplying (13) by P^T and P^{-T} respectively. Next we use the following lemma.

LEMMA 3.3. *Suppose $B = CC^T$ is positive definite. Then $Q := C^{-1}B^{1/2}$ is orthogonal.*

Proof. We see that $QQ^T = C^{-1}BC^{-T} = C^{-1}CC^TC^{-T} = I$. $\square$

THEOREM 3.4. *Suppose P is defined from X and S in such a way that if X and S are transformed into $\bar{X} := U^{-1}XU^{-T}$ and $\bar{S} := U^T S U$, then P is transformed into $\bar{P}$ with $\bar{P}^T \bar{P} = U^T P^T P U$. Suppose also that $H_P(XS)$ is positive definite. Then the direction that solves (16) is scale invariant. In particular this is true for the directions defined by $P = X^{-1/2}$, $P = S^{1/2}$, and $P = W^{-1/2}$.*

Proof. By letting $B := \bar{P}^T \bar{P} = U^T P^T P U$ and applying the lemma twice, we find that $\bar{P} = QPU$ for some orthogonal Q . It is also easy to see that $H_{\bar{P}}(\bar{X}\bar{S})$ is positive definite too, so that the original and the scaled system both have unique solutions by Theorem 3.1. From our remarks above, the resulting search direction in the scaled problem is the same as if we used $\bar{P} = PU$; but it is easy to check that the scaled direction $(\bar{\Delta X}, \Delta y, \bar{\Delta S}) := (U^{-1}\Delta XU^{-T}, \Delta y, U^T \Delta S U)$ satisfies the resulting system (16). As noted above the three given choices of P satisfy condition (a) of Theorem 3.2, and therefore the condition that $H_P(XS)$ is positive definite. $\square$

This theorem implies that the two H..K..M directions as well as the direction corresponding to $P = W^{-1/2}$ are scale invariant. In the next section we will show that this last direction is actually the NT direction. Table 1 summarizes the symmetry properties of the aforementioned directions.

3.4. The NT direction as a Newton direction. In this section, we show that the system (3) defining the NT direction is actually equivalent to (16) for some choices of P . These systems are not quite comparable: (16) allows for infeasible iterates, while (3) assumes that feasibility is maintained. Let us denote by (3') the system (3) where the two zeros on the right-hand side are replaced by r_p and R_d , respectively. We note that, independently of our work, Sturm and S. Zhang [20] derived the NT direction as a Newton direction for path-following algorithms using the symmetric transformation given in (8).

We first note that if an invertible matrix P satisfies $P^T P = W^{-1}$, then

$$(25) \quad P^T P X S P^{-1} P^{-T} = W^{-1} X S W = S W S W = S W W^{-1} X = S X.$$

Above we used the fact that

$$(26) \quad W^{-1} X = S W,$$

which follows from (6). Now, (25) indicates that

$$PXS P^{-1} = P^{-T}SXP^T.$$

Hence, using Theorem 3.2, we conclude that $H_P(XS)$ is positive definite.

There are several matrices P that satisfy $P^T P = W^{-1}$, for example $P = W^{-1/2}$ and $P = (X^{1/2}S X^{1/2})^{1/4}X^{-1/2}$. We will give another useful choice in the next section. We are now ready to show that the NT direction is a Newton direction.

THEOREM 3.5. *Let P be an invertible matrix such that $P^T P = W^{-1}$. Then the solutions to (16) and (3') are identical.*

Proof. It suffices to show the equivalence of (13) and the last equation in (3). We observe that (13) can be rewritten as

$$(27) \quad W^{-1}(X \Delta S + \Delta X S)W + \Delta S X + S \Delta X = 2\sigma\mu I - 2SX$$

by premultiplying it with P^T , postmultiplying it with P^{-T} , and using $P^T P = W^{-1}$. On the other hand, the last equation of (3) is equivalent to each of the two equations

$$(28) \quad W^{-1} \Delta X W^{-1} X + \Delta S X = \sigma\mu I - SX,$$

$$(29) \quad S \Delta X + S W \Delta S W = \sigma\mu S W X^{-1} W - S W S W.$$

For the first equality above, we postmultiplied the last equation of (3) by X , and for the second, we premultiplied it with SW and postmultiplied it with W . Now, adding up these two equations, and using (26) as well as its equivalent variants $SWX^{-1}W = I$, $WSW = X$, we get (27). This proves that the solution to (3') will also satisfy (16). However, (16) has a unique solution as indicated by the remarks above and Theorem 3.1, and therefore these two systems are equivalent. $\square$

4. Implementation. This section describes in detail how to perform the necessary computations at each iteration, and how to define and compute a second-order corrector modification (based on Mehrotra's corrector for LP [12]) of the NT direction.

4.1. Computation of W . The main purpose here is to show that W can be computed using two Cholesky factorizations and one SVD.

Let the Cholesky factorizations of the positive definite matrices X and S be

$$(30) \quad X = LL^T, \quad S = RR^T,$$

and let $UDV^T = R^T L$ be the SVD of $R^T L$. Define $Q := L^{-1}X^{1/2}$. By Lemma 3.3, Q is an orthogonal matrix. It is easily seen that

$$X^{1/2}S X^{1/2} = Q^T(L^T R)(R^T L)Q = (Q^T V)D^2(V^T Q).$$

Note that VD^2V^T is the eigenvalue decomposition of $L^T R R^T L = L^T S L$, so we could use this instead of the SVD. But the condition number of this matrix is the square of that of $R^T L$, so this may lead to a less stable algorithm. Since $Q^T V$ is orthogonal, we have

$$(31) \quad (X^{1/2}S X^{1/2})^{-1/2} = (Q^T V)D^{-1}(V^T Q).$$

From (31), W can be computed easily:

$$(32) \quad W = LVD^{-1}V^T L^T = GG^T,$$

where

$$(33) \quad G := LVD^{-1/2}.$$

We will use the matrix G in the computation of the direction $(\Delta X, \Delta y, \Delta S)$. If we wish to use the second-order corrector modification, we will also need $G^{-1} = D^{1/2}V^T L^{-1}$, but this is easy to compute. For later use, note the equation

$$(34) \quad G^T S G = G^{-1} X G^{-T} = D;$$

i.e., G scales X and S to the same diagonal matrix (compare (8)).

We remark that $G^{-T}G^{-1} = W^{-1}$ from (32), so that $P := G^{-1}$ satisfies $P^T P = W^{-1}$ and hence is yet another choice for P that yields the NT direction. In this case, not only do $P^{-T}SP^{-1}$ and PXP^T commute (see Theorem 3.2), they are actually the same diagonal matrix.

Note that, if we use the “reverse” Cholesky factorization of S , so that L is lower- but R upper-triangular in (30), then we save work in computing the lower-triangular matrix $R^T L$. Also, we do not need to keep the orthogonal matrix U ; V and D are all we need.

4.2. Computation of the NT direction. Using Theorem 3.5 and the choice $P = G^{-1}$, we can obtain the NT direction as the solution to

$$(35) \quad \begin{pmatrix} 0 & \mathcal{A} & 0 \\ \mathcal{A}^T & 0 & \mathcal{I} \\ 0 & E & F \end{pmatrix} \begin{pmatrix} \Delta y \\ \text{svec}(\Delta X) \\ \text{svec}(\Delta S) \end{pmatrix} = \begin{pmatrix} r_p \\ \text{svec}(R_d) \\ \text{svec}(R_c) \end{pmatrix},$$

where

$$E = G^{-1} \otimes_s G^T S, \quad F = G^{-1} X \otimes_s G^T, \\ r_p := b - \mathcal{A} \text{svec}(X), \quad R_d := C - S - \sum_{i=1}^m y_i A_i, \quad R_c := \sigma \mu I - D^2.$$

(We used $H_{G^{-1}}(XS) = (G^{-1}XSG + G^T SXG^{-T})/2 = D^2$ from (34).) Here, $\mathcal{A}$ is the matrix defined by (19).

The system (35) involves $m + n(n+1)$ linear equations. A standard method to solve this large system efficiently is to use block Gaussian elimination to reduce it to a Schur complement equation involving only Δy , given by

$$(36) \quad (\mathcal{A}E^{-1}F\mathcal{A}^T) \Delta y = r_p + \mathcal{A}E^{-1}F \text{svec}(R_d) - \mathcal{A}E^{-1} \text{svec}(R_c).$$

In our case, E^{-1} is not easy to obtain, but we need only $E^{-1}F$ and $E^{-1} \text{svec}(R_c)$. It is easy to check directly, by premultiplying by E , that

$$(37) \quad E^{-1}F = W \otimes_s W, \quad E^{-1} \text{svec}(R_c) = \text{svec}(\sigma \mu S^{-1} - X).$$

So $E^{-1}F$ is symmetric and positive definite, and it can be factorized as $E^{-1}F = KK^T$, where $K = G \otimes_s G$, obtained from the identity $W \otimes_s W = (G \otimes_s G)(G \otimes_s G)^T$. We will call such a factorization of $E^{-1}F$ a Cholesky-like factorization since the factors are transposes of one another, although they may not be triangular.

Using the Cholesky-like factorization of $E^{-1}F$, we have

$$\mathcal{A}E^{-1}F\mathcal{A}^T = \mathcal{B}\mathcal{B}^T,$$

and (36) can be rewritten as

$$(38) \quad \mathcal{B} \mathcal{B}^T \Delta y = r_p + \mathcal{B} h,$$

where

$$(39) \quad \mathcal{B}^T := K^T \mathcal{A}^T = [\text{svec}(G^T A_1 G) \cdots \text{svec}(G^T A_m G)]$$

and

$$(40) \quad \begin{aligned} h &= K^T \text{svec}(R_d) - K^{-1} E^{-1} \text{svec}(R_c) \\ &= \text{svec}(G^T R_d G - \sigma \mu D^{-1} + D). \end{aligned}$$

For the last expression, we used (37) and (34) to obtain

$$K^{-1} E^{-1} \text{svec}(R_c) = (G^{-1} \otimes_s G^{-1}) \text{svec}(\sigma \mu S^{-1} - X) = \text{svec}(\sigma \mu D^{-1} - D).$$

Note that (38) actually consists of the normal equations of a linear least-squares problem. To see this, choose a matrix $X_r \in S\mathbb{R}^{n \times n}$ such that $\mathcal{A} \text{svec}(X_r) = r_p$ (such a matrix exists because $\mathcal{A}$ has full row rank). By using $r_p = \mathcal{B} K^{-1} \text{svec}(X_r)$, we can rewrite (38) as

$$\mathcal{B} \mathcal{B}^T \Delta y = \mathcal{B} (K^{-1} \text{svec}(X_r) + h),$$

the normal equations of the least-squares problem

$$\min_{\Delta y} \|\mathcal{B}^T \Delta y - K^{-1} \text{svec}(X_r) - h\|.$$

There are two ways to compute Δy . One is to compute Δy from the Schur complement equation (36) via a Cholesky factorization of $\mathcal{A} E^{-1} F \mathcal{A}^T$. The other is to compute Δy from (38) via a QR factorization of $\mathcal{B}^T$ [5]. In exact arithmetic, these two methods are the same. But in finite precision arithmetic, they may have different numerical stability. Let κ be the condition number of $\mathcal{A} E^{-1} F \mathcal{A}^T$. It is well known from linear perturbation theory that solving (36) via a Cholesky factorization of $\mathcal{A} E^{-1} F \mathcal{A}^T$ produces a computed Δy whose relative error is dependent on κ , whereas solving (38) via a QR factorization of $\mathcal{B}^T$ produces a computed Δy whose relative error is dependent on a number that is between $\sqrt{\kappa}$ and κ . Thus solving (38) via the QR approach is conceivably better than using the Schur complement approach. In our experiments later, we consider both approaches.

Assuming that the step Δy has been computed from (38) via either QR or Cholesky factorization, we can compute ΔS simply from the second block equation of (35), namely,

$$(41) \quad \Delta S = R_d - \text{smat}(\mathcal{A}^T \Delta y),$$

and ΔX from the third block equation in (35), namely,

$$(42) \quad \Delta X = \text{smat}[E^{-1} \text{svec}(R_c) - E^{-1} F \text{svec}(\Delta S)].$$

We have shown that Δy in the NT direction can be computed via a QR factorization of $\mathcal{B}^T$. The same can be done for the H..K..M direction corresponding to

$P = S^{1/2}$, where now $E^{-1}F = X \otimes_s S^{-1}$. To show this, as before, the main task is getting a Cholesky-like factorization of $E^{-1}F$. By using (34), we have

$$(43) \quad \begin{aligned} E^{-1}F &= X \otimes_s S^{-1} = GDG^T \otimes_s GD^{-1}G^T \\ &= (G \otimes_s G)(D \otimes_s D^{-1})(G^T \otimes_s G^T). \end{aligned}$$

The middle term $D \otimes_s D^{-1}$ in (43) is a diagonal matrix with positive entries of the form $(d_i d_j^{-1} + d_j d_i^{-1})/2$, where the d_i 's are the diagonal entries of the matrix D . So the square root of $D \otimes_s D^{-1}$, say M , exists and can be computed trivially. It is now easy to see that we have the Cholesky-like factorization $E^{-1}F = KK^T$, where $K := (G \otimes_s G)M$. Y. Zhang [24] also observed the same Cholesky-like factorization of $E^{-1}F$ for the H.K.M direction, and Kojima [9] noted this for the linear system involving Kronecker (rather than symmetric Kronecker) products where ΔX is explicitly symmetrized afterwards.

In fact, the QR approach can be applied almost as easily to the direction that solves (16) whenever $E^{-1}F$ is symmetric. This is because $E^{-1}F$ can be expressed in the form

$$E^{-1}F = (P^{-1}Q \otimes_s P^{-1}Q) \tilde{D} (P^{-1}Q \otimes_s P^{-1}Q)^T,$$

where $\tilde{D}$ is a positive definite diagonal matrix and Q is an orthogonal matrix whose columns form an orthonormal basis that simultaneously diagonalizes PXP^T and $P^{-T}SP^{-1}$. Note that such a basis exists because by Theorem 3.2, the latter two matrices commute if $E^{-1}F$ is symmetric. (If $P = G^{-1}$, so that we have the NT direction, then Q and $\tilde{D}$ are identity matrices.)

Finally, we note that Δy in the AHO direction cannot be computed via the QR approach because the corresponding matrix $E^{-1}F$ is not symmetric.

4.3. A second-order correction to the NT direction. To derive a second-order correction to the NT direction, we suppose that we have approximations δX and δS to the search directions, and substitute them on the right-hand side of (12c'). Then, we get

$$E \operatorname{svec}(\Delta X) + F \operatorname{svec}(\Delta S) = \operatorname{svec}(R_s),$$

where $E = P \otimes_s P^{-T}S$, $F = PX \otimes_s P^{-T}$, and

$$R_s = \sigma\mu I - H_P(XS + \delta X \delta S)$$

(subscript s for second-order). Then

$$R_s = R_c + R_q,$$

where R_q is the “quadratic correction” $-H_P(\delta X \delta S)$. It is easily seen that ΔX and ΔS depend on P only through the product $P^T P$. For the NT direction we can therefore again choose $P = G^{-1}$.

If we then proceed exactly as in section 4.2, we see that we only have to replace $E^{-1}\operatorname{svec}(R_c)$ by $E^{-1}\operatorname{svec}(R_s)$ in (40) and (42). Thus, in the second expression for h in (40), $-\sigma\mu D^{-1} + D$ must be replaced by $-\sigma\mu D^{-1} + D - R_{NT}$, where the correction R_{NT} for the NT direction is given by

$$\operatorname{svec}(R_{NT}) := K^{-1}E^{-1}\operatorname{svec}[-H_{G^{-1}}(\delta X \delta S)].$$

Thus R_{NT} is the solution to $EK \text{svec}(R_{NT}) = \text{svec}[-H_{G^{-1}}(\delta X \delta S)]$. But $K = G \otimes_s G$ and $E = G^{-1} \otimes_s G^T S$, so this simplifies to the Lyapunov equation

$$(G^T S G) R_{NT} + R_{NT} (G^T S G) = -G^{-1}(\delta X \delta S) G - G^T(\delta S \delta X) G^{-T}.$$

But by (34), $G^T S G$ is the diagonal matrix D , and we can therefore write down the solution explicitly as

$$(44) \quad R_{NT} = -(G^{-1}(\delta X \delta S) G + G^T(\delta S \delta X) G^{-T}) ./ (de^T + ed^T),$$

where “./” means entrywise division, $d := \text{diag}(D)$, and $e := (1, \dots, 1)^T$. If we wished to use the Schur complement equation (36), the term $E^{-1} \text{svec}(R_c)$ on the right-hand side would be replaced by

$$(45) \quad E^{-1} \text{svec}(R_s) = \text{svec}(\sigma \mu S^{-1} - X) + (G \otimes_s G) \text{svec}(R_{NT}).$$

We also make this replacement in (42).

Note that the derivation of a second-order correction for the NT direction was possible only after the observation that the NT direction can be viewed as a Newton direction (see section 3.4); neither the original description (3) of the NT direction in [17, 18] nor the derivation by Sturm and S. Zhang [20] directly leads to a such a correction.

4.4. A Mehrotra predictor-corrector algorithm based on the NT direction. The results derived in the previous sections are the pieces involved in various steps of a Mehrotra-type predictor-corrector algorithm. Now we are ready to describe such a method based on the NT direction.

Remarks.

- In LP, the step-length parameter τ is fixed at a value very close to one, typically 0.99995. This choice seems to be too aggressive for Algorithm NT-PC-QR, and experiments have shown that $\tau = 0.98$ is usually a good parameter value. In our experiments, we also allow the step-length parameter τ to be chosen adaptively based on the step lengths taken in the previous iteration, i.e., by setting

$$(46) \quad \tau^{(k+1)} = 0.9 + 0.09 \min(\alpha^{(k)}, \beta^{(k)}),$$

where $\alpha^{(k)}$ and $\beta^{(k)}$ are the primal and dual step lengths in the k th corrector step, respectively.

- In our experiments with Algorithm NT-PC-QR, the parameter *expon* used for choosing σ in (47) is chosen to be one. This is more conservative than using an exponent of three as in LP.

ALGORITHM NT-PC-QR. Suppose we are given an initial iterate (X^0, y^0, S^0) with X^0, S^0 positive definite. Choose $\tau \in (0, 1)$ and $expon \in \{1, 2, 3\}$.

For $k = 0, 1, \dots$,

let the current and the next iterate be (X, y, S) and (X^+, y^+, S^+) , respectively. Let $\mu := X \bullet S/n$.

- (Predictor step)
Compute the Newton step $(\delta X, \delta y, \delta S)$ from (38), (41), and (42), with $\sigma := 0$ in (40).
- Determine the parameter σ :

$$(47) \quad \sigma := \left[\frac{(X + \alpha \delta X) \bullet (S + \beta \delta S)}{X \bullet S} \right]^{expon}.$$

Here

$$(48) \quad \alpha := \min \left(1, \frac{-\tau}{\lambda_{\min}(X^{-1} \delta X)} \right), \quad \beta := \min \left(1, \frac{-\tau}{\lambda_{\min}(S^{-1} \delta S)} \right)$$

are step lengths chosen to ensure that $X + \alpha \delta X$ and $S + \beta \delta S$ are positive definite (if the minimum eigenvalue in either expression is positive, we ignore the corresponding term).

- (Corrector step)
With σ and μ determined as above, compute the Newton step $(\Delta X, \Delta y, \Delta S)$ from (38), (41), and (42) with the matrix $-\sigma \mu D^{-1} + D$ in (40) replaced by the matrix $-\sigma \mu D^{-1} + D - R_{NT}$, and the vector $E^{-1} \text{svec}(R_c)$ in (42) replaced by $E^{-1} \text{svec}(R_s)$ in (45), where R_{NT} is given by (44).
- Update (X, y, S) to (X^+, y^+, S^+) by

$$X^+ = X + \alpha \Delta X, \quad y^+ = y + \beta \Delta y, \quad S^+ = S + \beta \Delta S,$$

where α and β are determined from (48) with $\delta X, \delta S$ replaced by $\Delta X, \Delta S$, so that X^+ and S^+ are positive definite.

- When solving (38) in the predictor step by Householder QR factorization, the Householder vectors are saved for later use in the corrector step, as is normally done for the LU factorization of the Schur complement matrix. In this way, the corrector step can be computed with just $O(mn^2)$ flops.
- In computing α in (48), it is cheaper to compute the minimum eigenvalue of the symmetric matrix $L^{-1} \delta X L^{-T}$, where $X = LL^T$ is the Cholesky factorization of X . This matrix has the same spectrum as that of the nonsymmetric matrix $X^{-1} \delta X$. We do the same when computing β .
- Actually, it is more efficient to do most of the computations with “ G ”-scaled quantities. At the beginning of the iteration, we compute G and hence $\mathcal{B}$ and $\bar{R}_d := G^T R_d G$. Hence we calculate h from (40) and thus δy from (38). We then obtain

$$\overline{\delta S} := G^T \delta S G = \bar{R}_d - \text{smat}(\mathcal{B}^T \delta y)$$

and

$$\begin{aligned} \overline{\delta X} &:= G^{-1} \delta X G^{-T} = \text{smat}(K^{-1} E^{-1} \text{svec}(R_c)) - \overline{\delta S} \\ &= \sigma \mu D^{-1} - D - \bar{R}_d + \text{smat}(\mathcal{B}^T \delta y). \end{aligned}$$

Next we compute α and β from (48), but using $\lambda_{\min}(D^{-1/2} \bar{\delta X} D^{-1/2})$, and similarly for S ; note that in transformed space we want $D + \alpha \bar{\delta X} \succ 0$, since X and S both transform to $\bar{X} = \bar{S} = D$. Then we obtain σ from (47), but using barred quantities. For the corrector step, $-\sigma \mu D^{-1} + D$ in (40) is replaced by

$$-\sigma \mu D^{-1} + D + (\bar{\delta X} \bar{\delta S} + \bar{\delta S} \bar{\delta X}) / (de^T + ed^T).$$

Then proceed as above to obtain Δy , $\bar{\Delta X}$, and $\bar{\Delta S}$. Calculate new step lengths α and β . Finally we set

$$X^+ = G(D + \alpha \bar{\Delta X})G^T, \quad y^+ = y + \beta \Delta y, \quad S^+ = S + \beta \Delta S,$$

where ΔS is computed from (41) to maintain dual feasibility.

- The stopping criterion we use in our algorithm is as follows. We stop the iteration if $\sigma > 1$. This usually happens when the duality gap is in the range of the achievable accuracy of the algorithm. However, we note that more precise stopping criteria need to be devised. We also stop the iteration if the step lengths α and β in the corrector step are smaller than 10^{-6} .
- Assuming that the Householder vectors are saved when solving (38) in the predictor step, the complexity of each iteration of this algorithm is $3mn^3 + \frac{1}{2}m^2n^2 + O(n^3 + mn^2)$ floating point operations.

5. Numerical experiments. Our purpose here is to compare Algorithm NT-PC-QR with methods using the same algorithmic framework, but computing the directions from the Schur complement equation (36) via a Cholesky or an LU factorization of $\mathcal{A}E^{-1}F\mathcal{A}^T$. In the latter methods, we consider three different search directions (we only consider one of the H.K.M directions, the one that coincides with the direction of Helmberg et al. [7] and the one that is easier to compute—see Y. Zhang [23]):

1. AHO direction ($P = I$);
2. H.K.M direction ($P = S^{\frac{1}{2}}$);
3. NT direction ($P = G^{-1}$).

These three algorithms will be referred to as Algorithm AHO-PC-Sch, H.K.M-PC-Sch, and NT-PC-Sch, respectively.

Our code is based on one of Alizadeh, Haeberly, and Overton [3] for feasible iterates; we are grateful to them for making it available to us. We also mention here that they have had success with the larger step-length parameter value of 0.999.

We have already discussed how $(\Delta X, \Delta y, \Delta S)$ is computed for Algorithm NT-PC-QR. Now we briefly describe how it is computed for the other three methods. For simplicity, we only discuss the predictor step. The step δy is first computed from the Schur complement equation (36) via a Cholesky or LU factorization of the $m \times m$ Schur complement matrix $\mathcal{A}E^{-1}F\mathcal{A}^T$, according to whether the matrix is known to be symmetric positive definite or not. Once δy is obtained, δS is computed from (41) and δX from (42). It is clear that the main work involved in these methods is the computation of the coefficient matrix and the right-hand side vector of the Schur complement equation. In Table 2, we summarize some of the formulae that are needed in this computation. The complexity of each iteration of these methods is also included, ignoring lower-order terms like $O(n^3 + mn^2)$.

For all the problems below, the starting points are feasible and so feasibility is maintained throughout in exact arithmetic. Our formulae for ΔS guarantee that

TABLE 2

Summary of the formulae involved in the computation of the coefficient matrix and the right-hand side of the Schur complement equation for the AHO, H..K..M, and NT directions. We count one addition and one multiplication each as one flop.

Directions	AHO ($P = I$)	H..K..M ($P = S^{1/2}$)	NT ($P = G^{-1}$)
E	$I \otimes_s S$	$S^{1/2} \otimes_s S^{1/2}$	$G^{-1} \otimes_s G^T S$
F	$X \otimes_s I$	$S^{1/2} X \otimes_s S^{-1/2}$	$G^{-1} X \otimes_s G^T$
$E^{-1}F$	$(I \otimes_s S)^{-1}(X \otimes_s I)$	$X \otimes_s S^{-1}$	$W \otimes_s W$
R_c	$\sigma\mu I - (XS + SX)/2$	$\sigma\mu I - S^{1/2} X S^{1/2}$	$\sigma\mu I - D^2$
R_q	$-(\delta X \delta S + \delta S \delta X)/2$	$-H_{S^{1/2}}(\delta X \delta S)$	see (45)
Complexity	$8mn^3 + m^2n^2$	$4mn^3 + 0.5m^2n^2$	$3mn^3 + 0.5m^2n^2$

dual feasibility is preserved to high accuracy, but the same is not true for primal feasibility. Hence we adjust $\text{svec}(\Delta X)$ (before updating to X^+) in each iteration to regain feasibility by projection onto the null space of $\mathcal{A}$, i.e., replacing $\text{svec}(\Delta X)$ by $[I - \mathcal{A}^T(\mathcal{A}\mathcal{A}^T)^{-1}\mathcal{A}]\text{svec}(\Delta X)$. This step is inexpensive. To compute the projection, we only need to compute the Cholesky factorization of the $m \times m$ matrix $\mathcal{A}\mathcal{A}^T$ once at the beginning of the algorithm. (We use the Cholesky factorization here instead of a QR factorization to avoid using the `smat` and `svec` operations, which are expensive in MATLAB, in the Schur complement methods; for Algorithm NT-PC-QR an alternative would be to use the QR factorization of $K^T\mathcal{A}^T$ obtained in the first iteration to do the primal feasibility correction in all iterations.)

We compare the performance of Algorithm NT-PC-QR and the three methods just mentioned on the following classes of semidefinite programming problems.

- (1) *Random SDP*: The starting point does not necessarily lie on the central path.
- (2) *Norm minimization problem*:

$$(49) \quad \min_{x \in \mathbb{R}^m} \|A_0 + \sum_{k=1}^m x_k A_k\|,$$

where the A_k , $k = 0, \dots, m$, are real $N \times N$ matrices and the norm is the matrix 2-norm. It is well known that this problem can be expressed as an SDP problem involving $m + 2$ symmetric matrices of dimension $n \times n$, where $n = 2N$ [22]. For this problem, a feasible starting point is readily available, and we choose (X_0, y_0, S_0) to be feasible, but not necessarily lying on the central path.

- (3) *Chebyshev approximation problem for a matrix*:

$$(50) \quad \min_p \|p(A)\|,$$

where the minimization is over the class of monic polynomials of degree m and the norm is the matrix 2-norm. Here A is a real $N \times N$ matrix. The Chebyshev problem (50) is a special case of the norm minimization problem and thus can be expressed as an SDP problem.

We should note that since the power basis $\{I, A, \dots, A^m\}$ is highly ill conditioned in general, it should be replaced by a better-conditioned alternative for numerical stability. In our actual computation, we use the orthonormal basis $\{Q_1, \dots, Q_{m+1}\}$, with respect to the inner product $G \bullet K$, obtained from the power basis via a modified

Gram–Schmidt procedure. For details on how this change of basis transforms the original problem, see [21]. Again, we choose a feasible starting point for this problem.

(4) *Max-Cut problem*:

$$(51) \quad \begin{aligned} & \min L \bullet X \\ & \text{s.t.} \quad \text{diag}(X) = \frac{e}{4}, \quad X \succeq 0, \end{aligned}$$

where $L = A - \text{diag}(Ae)$, e is the vector of all ones, and A is the weighted adjacency matrix of a graph [7]. In our experiments, we consider only unweighted graphs where each edge is present independently with probability one half. We choose the feasible starting point

$$X_0 = \text{diag}\left(\frac{e}{4}\right), \quad y_0 = -1.1 \text{abs}(L)e, \quad S_0 = L - \text{diag}(y_0),$$

where $\text{abs}(L)$ denotes the matrix of absolute values of the entries of L .

(5) *Educational testing problem (ETP)*:

$$(52) \quad \begin{aligned} & \max e^T d \\ & \text{s.t.} \quad A - \text{diag}(d) \succeq 0, \quad d \geq 0, \end{aligned}$$

where A is an $N \times N$ positive definite matrix. This problem can readily be expressed as an SDP problem of the form (2), involving symmetric matrices of dimension $n \times n$, where $n = 2N$. Again, a feasible starting point is chosen for this problem.

In our experiments, all the computations were performed in MATLAB [13]. For each class of the SDP problems mentioned above, we solved 10 instances with random data. That is, the given matrices are random, with entries chosen from the normal distribution with zero mean and unit variance. For the ETP, A is the matrix product of such a random matrix and its transpose so that the resulting matrix is symmetric positive definite.

For each set of 10 instances, we compare the algorithms using the adaptive step-length strategy given in (46) and also use a fixed step-length parameter with $\tau = 0.98$. The parameter *expon* used in our experiments is given by

$$\text{expon} = \begin{cases} 3 & \text{for the AHO method,} \\ 1 & \text{for the H..K..M method,} \\ 1 & \text{for the NT methods.} \end{cases}$$

For each class, we plot the convergence curves of the duality gap of one of the instances. The plots are shown in Figures 1–5. The dashed curve corresponds to the AHO direction, the dotted curve corresponds to the H..K..M direction, and the solid curves correspond to the NT direction. For the most part, the two solid curves coincide with one another, indicating that the performances of Algorithm NT-PC-QR and NT-PC-Sch are about the same on most problems. This also shows that for the NT direction, solving the Schur complement equation by Cholesky factorization is as good as solving by QR factorization. Since the behavior of these two algorithms depends mainly on the properties of the NT direction rather than on how the direction is computed, we will not distinguish them in the subsequent discussion, but refer to them collectively as the NT methods.

In Tables 3 and 4, corresponding to the adaptive step-length parameter given in (46) and the fixed step-length parameter of 0.98, respectively, we give the average

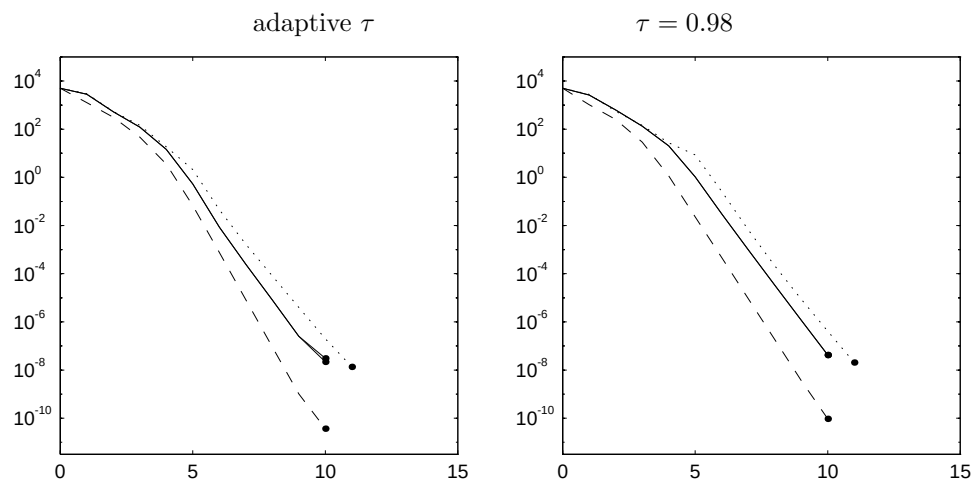


FIG. 1. Convergence curves of the duality gaps of a random SDP problem for different methods. The dashed curve corresponds to the AHO direction, the dotted curve corresponds to the H..K..M direction, and the solid curves correspond to the NT direction.

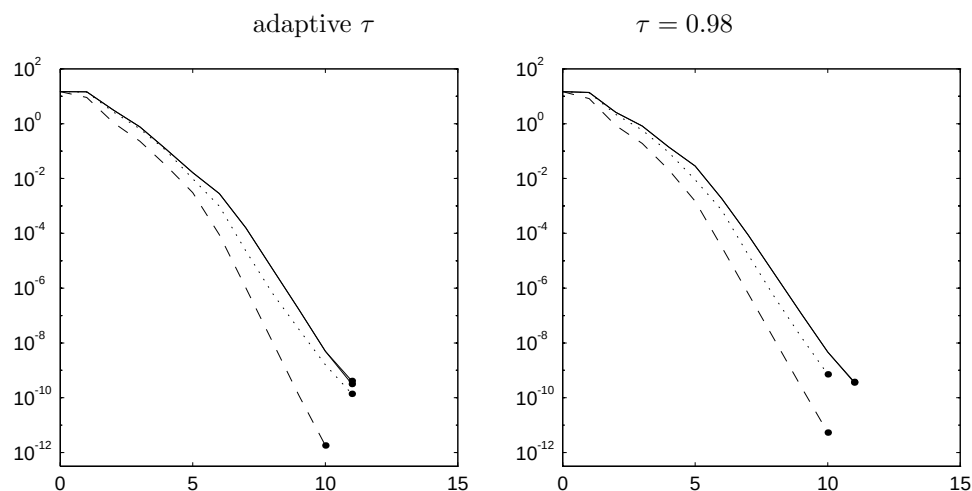


FIG. 2. Same as Figure 1 but for a norm-minimization problem.

number of iterations and CPU time for each method to reduce the duality gap by a factor of 10^{10} . (Note that this is much better than a factor of $\epsilon_{\text{mach}}^{-1/2}$, which informal reasoning might suggest as the limit for methods based on the Schur complement approach.) The times given exhibit mild fluctuations due to other jobs being present on the machines being used, but the trends are clear.

A summary of our observations is given below.

1. For Algorithm AHO-PC-Sch, the achievable accuracy in the duality gap is typically 10 to 100 times smaller than that obtained from methods based on the H..K..M and NT directions. For the first four classes of SDP problems we consider here, the duality gap can usually reach a level smaller than 10^{-10} to 10^{-12} for Algorithm AHO-PC-Sch. It is surprising that AHO-PC-Sch can achieve such an accuracy in the duality gap despite the fact that the

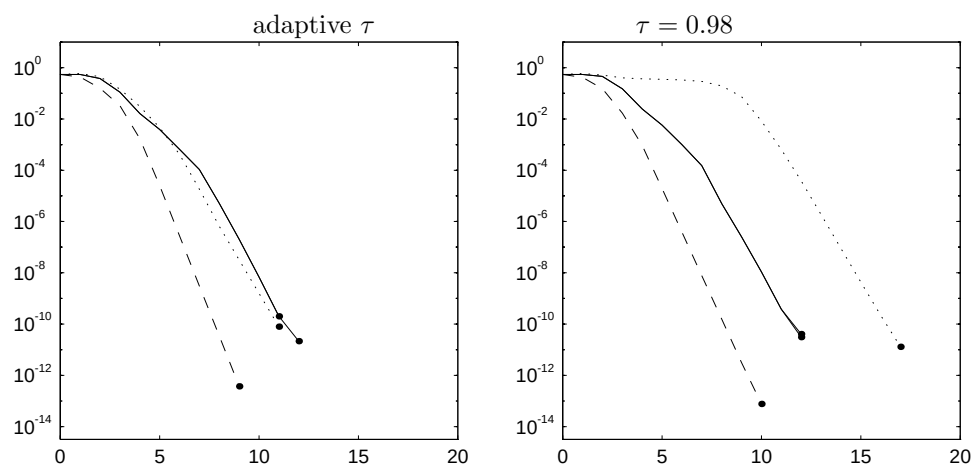


FIG. 3. Same as Figure 1 but for a Chebyshev problem.

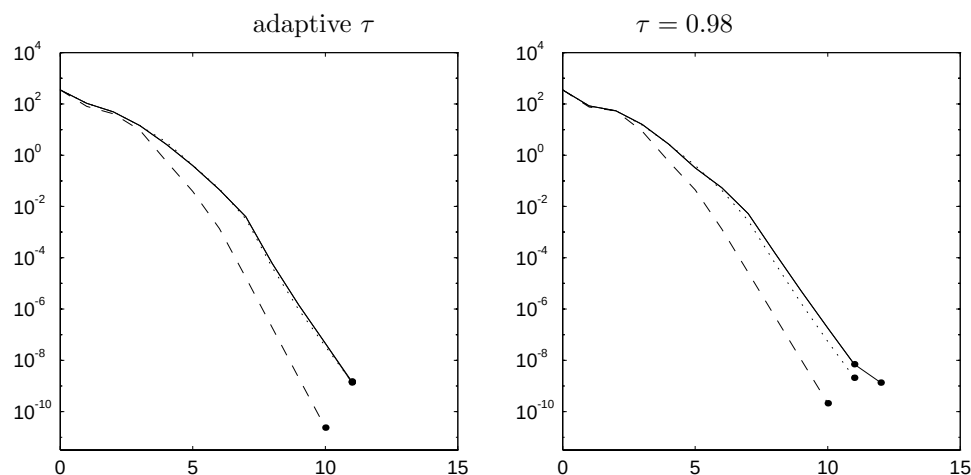


FIG. 4. Same as Figure 1 but for a Max-Cut problem.

condition number of the Schur complement matrix involved is of the order of the reciprocal of the duality gap.

2. Besides being the most accurate, Algorithm AHO-PC-Sch is also the most efficient (in terms of the number of iterations) on the sets of problems we consider here, but the NT methods are almost as efficient, typically requiring only two or three more iterations to achieve a given precision in the duality gap. The H.K.M method is also almost as efficient as the AHO method, except on the ETPs.
3. However, each iteration of Algorithm AHO-PC-Sch is about twice as expensive as each iteration of the H.K.M or the NT methods. Among the latter three methods, each iteration of NT-PC-QR is slightly more expensive than those of NT-PC-Sch (but not always), which in turn is slightly more expensive than each iteration of H.K.M-PC-Sch. Consequently, overall the

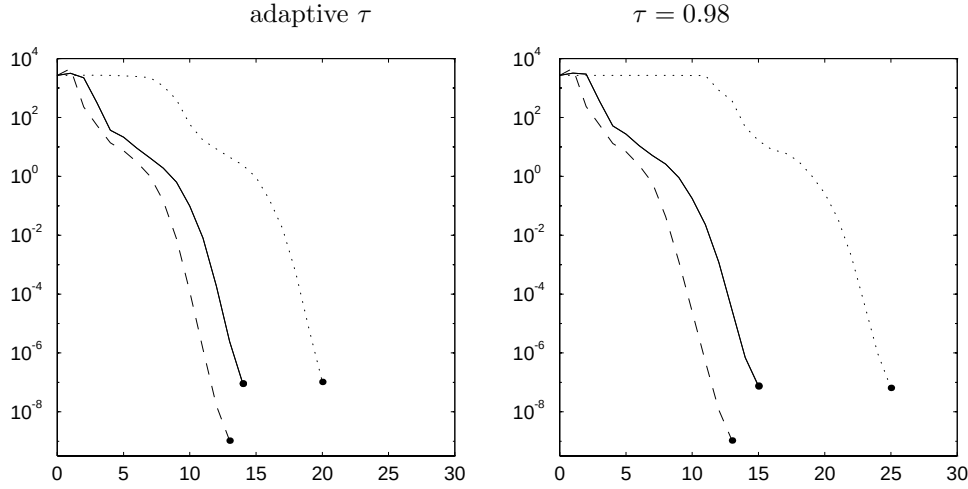


FIG. 5. Same as Figure 1 but for an ETP. This class of SDP problems proves to be more difficult for the H..K..M direction.

H..K..M method takes the least average amount of CPU time to reduce the duality gap by 10^{10} , the NT methods follow closely, and the AHO method is the slowest.

4. Though the NT methods are not the best in achieving high accuracy, they may be the most robust in the sense that problems of stagnation such as using very small step lengths seem less likely to occur. We have not observed stagnation problems with the NT methods in our numerical experiments, while the AHO method occasionally stagnates on the Chebyshev approximation problems for matrices, and the H..K..M method often encounters such difficulties on the ETPs. This difficulty does not seem to be due to the Schur complement approach; using a QR implementation of the H..K..M method yields similar results.
5. Choosing an adaptive step-length parameter can alleviate stagnation problems that are encountered from using a fixed step-length parameter. The improvement is especially noticable for the H..K..M method in solving the ETPs.

As we already mentioned in the introduction, the good performance of the methods that are based on the Schur complement approach was observed after the release of the first version of this paper, in which NT-PC-QR had been observed to be superior to the rest. We note that these improvements in the Schur complement-based methods come from careful implementations; for example, the code now symmetrizes explicitly quantities that ought to be symmetric in exact arithmetic.

Appendix. We present some useful properties of the standard and symmetrized Kronecker products.

- Standard Kronecker product:
 1. $G \otimes K = [g_{ij}K]$.
 2. $(G \otimes K)\text{vec}(H) = \text{vec}(KHG^T)$ (where vec is defined below).
 3. $(G \otimes K)^T = G^T \otimes K^T$.
 4. $G \otimes I$ is symmetric iff G is.
 5. $(G \otimes K)^{-1} = G^{-1} \otimes K^{-1}$.

TABLE 3
Computational results on different classes of SDP problems. Ten instances with random data are considered for each class of the SDP problems. The computations were done on a DEC AlphaStation 500 (300 MHz). An adaptive step-length parameter choice is used in the algorithms.

Adaptive τ Examples	Average number of iterations to reduce the duality gap by 10^{10}				Average CPU time (secs.) to reduce the duality gap by 10^{10}			
	AHO	H..K..M	NT	NT(QR)	AHO	H..K..M	NT	NT(QR)
Random SDP $n = 100$ $m = 25$	9.2	10.2	9.8	9.8	30.7	22.4	20.6	22.7
Norm minimization $n = 100$ $m = 25$	9.0	11.0	11.2	11.2	90.8	59.8	67.1	71.5
Chebyshev $n = 100$ $m = 25$	8.7	11.0	11.1	11.1	89.0	60.2	66.5	70.8
Max-Cut $n = 50$ $m = 50$	9.3	11.0	11.0	11.0	22.7	13.9	14.0	20.1
ETP $n = 100$ $m = 50$	13.4	20.2	15.8	15.8	70.8	45.3	38.7	44.2

TABLE 4
Same as Table 3 but for fixed step-length parameter with $\tau = 0.98$.

$\tau = 0.98$ Examples		Average number of iterations to reduce the duality gap by 10^{10}				Average CPU time (secs.) to reduce the duality gap by 10^{10}			
		AHO	H..K..M	NT	NT(QR)	AHO	H..K..M	NT	NT(QR)
Random SDP	$n = 100$ $m = 25$	9.0	10.4	10.1	10.1	27.6	19.7	20.2	22.3
	Norm minimization	9.3	11.0	11.5	11.4	90.4	58.2	66.1	72.9
Chebyshev	$n = 100$ $m = 25$	9.4*	13.0	11.4	11.5	90.3*	68.2	66.4	72.7
	Max-Cut	9.4	11.1	11.1	11.1	22.3	13.2	14.0	19.3
ETP	$n = 100$ $m = 50$	13.7	25.5	17.2	17.1	65.7	57.1	41.6	51.0

*The AHO method fails on one of the instances due to step lengths going below 10^{-6} .
The number reported here does not include this unsuccessful instance.

6. $(G \otimes K)(H \otimes L) = GH \otimes KL$.
7. If $\Lambda(G) = \{\lambda_i\}$ and $\Lambda(K) = \{\mu_j\}$, then $\Lambda(G \otimes K) = \{\lambda_i \mu_j\}$. If x_i and y_j are the eigenvectors corresponding to the eigenvalues λ_i and μ_j of G and K , then $\text{vec}(y_j x_i^T)$ is the eigenvector corresponding to the eigenvalue $\lambda_i \mu_j$ of $G \otimes K$.
8. $\text{vec}(G)^T \text{vec}(K) = G \bullet K$.
- Symmetrized Kronecker product:
 1. $(G \otimes_s K) \text{svec}(H) = \frac{1}{2} \text{svec}(KHG^T + GHK^T)$.
 2. $G \otimes_s K = K \otimes_s G$.
 3. $(G \otimes_s K)^T = G^T \otimes_s K^T$.
 4. $G \otimes_s I$ is symmetric iff G is.
 5. $(G \otimes_s G)^{-1} = G^{-1} \otimes_s G^{-1}$.
 6. $(G \otimes_s K)(H \otimes_s L) = \frac{1}{2}(GH \otimes_s KL + GL \otimes_s KH)$.
 7. $(G \otimes_s K)(H \otimes_s H) = (GH \otimes_s KH)$ and $(H \otimes_s H)(G \otimes_s K) = (HG \otimes_s HK)$.
 8. If $\Lambda(G) = \{\lambda_i\}$, then $\Lambda(G \otimes_s G) = \{\lambda_i \lambda_j\}$. If x_i and x_j are the eigenvectors corresponding to the eigenvalues λ_i, λ_j of G , then $\text{svec}(\frac{1}{2}(x_i x_j^T + x_j x_i^T))$ is the eigenvector corresponding to the eigenvalue $\lambda_i \lambda_j$ of $G \otimes_s G$.
 9. Let G and K be commuting symmetric matrices with the common basis of eigenvectors $\{x_i\}$ and corresponding eigenvalues λ_i and μ_i , respectively. Then $\Lambda(G \otimes_s K) = \{\frac{1}{2}(\lambda_i \mu_j + \lambda_j \mu_i)\}$. Also, $\text{svec}(\frac{1}{2}(x_i x_j^T + x_j x_i^T))$ is an eigenvector corresponding to the eigenvalue $\frac{1}{2}(\lambda_i \mu_j + \lambda_j \mu_i)$ of $G \otimes_s K$ [3].
 10. $\text{svec}(G)^T \text{svec}(K) = G \bullet K$.
 11. If G and K are symmetric and positive definite, then so is $G \otimes_s K$.

The properties of the symmetrized Kronecker product can be verified by relating it to the standard Kronecker product, and then using its properties. To this end, consider the $n(n+1)/2 \times n^2$ matrix U such that $U \text{vec}(H) = \text{svec}(H)$ and $U^T \text{svec}(H) = \text{vec}(H)$ for all $n \times n$ symmetric matrices H . Here

$$\text{vec}(H) := (h_{11}, h_{21}, \dots, h_{n1}, h_{12}, \dots, h_{nn})^T.$$

If we label the rows of U in the order $(1, 1), (2, 1), \dots, (n, 1), (2, 2), (3, 2), \dots, (n, 2), (3, 3), \dots, (n, n)$ and its columns in the order $(1, 1), (2, 1), \dots, (n, 1), (1, 2), \dots, (n, 2), (1, 3), \dots, (n, n)$, then

$$U_{(i,j),(k,l)} = \begin{cases} 1 & \text{if } i = j = k = l, \\ 1/\sqrt{2} & \text{if } i = k \neq j = l, \text{ or } i = l \neq j = k, \\ 0 & \text{otherwise.} \end{cases}$$

For example, for $n = 3$,

$$U = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/\sqrt{2} & 0 & 1/\sqrt{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/\sqrt{2} & 0 & 0 & 0 & 1/\sqrt{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/\sqrt{2} & 0 & 1/\sqrt{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Note that $U U^T = I_{n(n+1)/2}$. Although $U^T U \neq I_{n^2}$, we have that $U^T U \text{vec}(H) = \text{vec}(H)$ for every $n \times n$ symmetric matrix H . In fact, $U^T U$ is the orthogonal projection matrix onto the space of symmetric matrices.

With the matrix U , $G \otimes_s K$ can be expressed in terms of the standard Kronecker products of G and K as follows:

$$G \otimes_s K = \frac{1}{2} U(G \otimes K + K \otimes G) U^T.$$

Some properties of the symmetrized Kronecker product can also be derived directly from the definition (15), which is listed as Property 1 above. For example, Property 6 holds because for any matrix M

$$\begin{aligned} (G \otimes_s K)(H \otimes_s L) \text{svec}(M) &= \frac{1}{2} (G \otimes_s K) \text{svec}(L M H^T + H M L^T) \\ &= \frac{1}{4} \text{svec}(K L M H^T G^T + G H M L^T K^T \\ &\quad + K H M L^T G^T + G L M H^T K^T) \\ &= \frac{1}{2} (G H \otimes_s K L + G L \otimes_s K H) \text{svec}(M). \end{aligned}$$

Since M is arbitrary, the property follows.

REFERENCES

- [1] F. ALIZADEH, *Combinatorial Optimization with Interior Point Methods and Semi-definite Matrices*, Ph.D. thesis, University of Minnesota, Minneapolis, MN, 1991.
- [2] F. ALIZADEH, *Semidefinite programming home page*, <http://rutcor.rutgers.edu/~alizadeh/sdp.html>.
- [3] F. ALIZADEH, J. A. HAEBERLY, AND M. OVERTON, *Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results*, SIAM J. Optim., 8 (1998), pp. 746–768.
- [4] T. ANDO, *Concavity of certain maps and positive definite matrices and applications to Hadamard products*, Linear Algebra Appl., 26 (1979), pp. 203–241.
- [5] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, 2nd ed., The Johns Hopkins University Press, Baltimore, MD, 1989.
- [6] C. HELMBERG, *Semidefinite programming home page*, <http://www.zib.de/helmberg/semidef.html>.
- [7] C. HELMBERG, F. RENDL, R. VANDERBEI, AND H. WOLKOWICZ, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.
- [8] R. HORN, *private communication*, University of Utah, Salt Lake City, UT, 1996.
- [9] M. KOJIMA, *private communication*, Tokyo Institute of Technology, Japan, 1996.
- [10] M. KOJIMA, S. SHINDOH, AND S. HARA, *Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.
- [11] C.-J. LIN AND R. SAIGAL, *A Predictor Corrector Method for Semi-definite Linear Programming*, manuscript, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI, 1995.
- [12] S. MEHROTRA, *On the implementation of a primal-dual interior point method*, SIAM J. Optim., 2 (1992), pp. 575–601.
- [13] THE MATHWORKS, INC., *MATLAB Reference Guide*, The MathWorks, Inc., Natick, MA, 1992.
- [14] R. D. C. MONTEIRO, *Primal-dual path-following algorithms for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 663–678.
- [15] Y. NESTEROV AND A. S. NEMIROVSKII, *Conic formulation of a convex programming problem and duality*, Optimization Methods and Software, 1 (1992), pp. 95–115.
- [16] Y. NESTEROV AND A. S. NEMIROVSKII, *Interior Point Polynomial Methods in Convex Programming: Theory and Algorithms*, SIAM, Philadelphia, PA, 1994.
- [17] Y. NESTEROV AND M. J. TODD, *Self-scaled barriers and interior-point methods in convex programming*, Math. Oper. Res., 22 (1997), pp. 1–42.
- [18] Y. NESTEROV AND M. J. TODD, *Primal-dual interior-point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.

- [19] M. SHIDA, S. SHINDOH, AND M. KOJIMA, *Existence of search directions in interior-point algorithms for the SDP and the monotone SDLCP*, SIAM J. Optim., 8 (1998), pp. 387–396.
- [20] J. F. STURM AND S. ZHANG, *Symmetric Primal-Dual Path Following Algorithms for Semidefinite Programming*, Tech. Report 9554/A, Econometric Institute, Erasmus University, Rotterdam, The Netherlands, 1995.
- [21] K. C. TOH AND L. N. TREFETHEN, *The Chebyshev polynomials of a matrix*, SIAM J. Matrix Anal. Appl., to appear.
- [22] L. VANDENBERGHE AND S. BOYD, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.
- [23] Y. ZHANG, *On extending some primal-dual interior-point algorithms from linear programming to semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 365–386.
- [24] Y. ZHANG, *private communication*, University of Maryland Baltimore County, Baltimore, MD, 1996.

POLYNOMIAL CONVERGENCE OF PRIMAL-DUAL ALGORITHMS FOR SEMIDEFINITE PROGRAMMING BASED ON THE MONTEIRO AND ZHANG FAMILY OF DIRECTIONS*

RENATO D. C. MONTEIRO†

Abstract. This paper establishes the polynomial convergence of the class of primal-dual feasible interior-point algorithms for semidefinite programming (SDP) based on the Monteiro and Zhang family of search directions. In contrast to Monteiro and Zhang's work [*Math. Programming*, 81 (1998), pp. 281–299], here no condition is imposed on the scaling matrix that determines the search direction. We show that the polynomial iteration-complexity bounds of two well-known algorithms for linear programming, namely the short-step path-following algorithm of Kojima, Mizuno, and Yoshise [*Math. Programming*, 44 (1989), pp. 1–26] and Monteiro and Adler [*Math. Programming*, 44 (1989), pp. 27–41 and pp. 43–66] and the predictor-corrector algorithm of Mizuno, Todd, and Ye [*Math. Oper. Res.*, 18 (1993), pp. 945–981] carry over into the context of SDP. Since the Monteiro and Zhang family of directions includes the Alizadeh, Haeberly, and Overton direction, we establish for the first time the polynomial convergence of algorithms based on this search direction.

Key words. semidefinite programming, interior-point methods, polynomial complexity, path-following methods, primal-dual methods

AMS subject classifications. 65K05, 90C25, 90C30

PII. S1052623496308618

1. Introduction. Several authors have discussed generalizations of interior-point algorithms for linear programming (LP) to the context of semidefinite programming (SDP). Nesterov and Nemirovskii propose in their landmark works [22, 23] a general approach for using interior-point methods to solve convex programs based on the notion of self-concordant functions. (See their book [25] for a comprehensive treatment of this subject.) They show that the problem of minimizing a linear function over a convex set can be solved in “polynomial time” as long as a self-concordant barrier function for the convex set is known. In particular, they show that linear programs, convex quadratic programs with convex quadratic constraints, and semidefinite programs all have explicit and easily computable self-concordant barrier functions, and hence can be solved in “polynomial time.” On the other hand, Alizadeh [1] extends Ye's projective potential reduction algorithm [33] for LP to SDP and argues that many known interior point algorithms for LP can also be transformed into algorithms for SDP in a mechanical way. Since then, many authors have proposed interior-point algorithms for solving SDP problems, including Alizadeh, Haeberly, and Overton [2, 3]; Freund [4]; Helmberg et al. [5]; Jarre [7]; Kojima, Shida, and Shindoh [10, 12]; Kojima, Shindoh, and Hara [13]; Lin and Saigal [14]; Luo, Sturm, and Zhang [15]; Monteiro [17]; Monteiro and Zhang [21]; Nesterov and Nemirovskii [24]; Nesterov and Todd [26, 27]; Potra and Sheng [28]; Sturm and Zhang [29]; Tseng [31]; Vandenberghe and Boyd [32]; and Zhang [34]. Most of these more recent works concentrate on primal-dual methods.

*Received by the editors August 28, 1996; accepted for publication (in revised form) July 24, 1997; published electronically July 20, 1998. This research was partially supported by National Science Foundation grants INT-9600343 and CCR-9700448 and Office of Naval Research grant N00014-94-1-0340.

<http://www.siam.org/journals/siopt/8-3/30861.html>

†School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332 (monteiro@isye.gatech.edu).

The first SDP algorithms that are extensions of primal-dual LP algorithms, such as the long-step path-following algorithm of Kojima, Mizuno, and Yoshise [9], the short-step path-following algorithm of Kojima, Mizuno, and Yoshise [8] and Monteiro and Adler [18, 19], and the predictor-corrector algorithm of Mizuno, Todd, and Ye [16], use one of the following three search directions: (i) the Alizadeh, Haeberly, and Overton (AHO) direction proposed in [2]; (ii) a direction independently proposed by Kojima, Shindoh, and Hara [13] and Helmberg et al. [5], and later rediscovered by Monteiro [17], which we refer to as the KSH/HRVW/M-direction; and (iii) the Nesterov and Todd (NT) direction introduced in [26, 27]. Application of the Newton method to the central path equation $XS = \sigma\mu I$ results in an equation of the form

$$(1) \quad X\Delta S + \Delta XS = \sigma\mu I - XS,$$

which in general yields nonsymmetric directions. The AHO-direction corresponds to the symmetric equation obtained by symmetrizing both sides of the previous equation.

Another way of symmetrizing (1) is to first apply a similarity transformation $P(\cdot)P^{-1}$ to both sides and then symmetrize it. Such an approach was first introduced by Monteiro [17] for the cases of $P = X^{-1/2}$ and $P = S^{1/2}$. The resulting directions were found to be equivalent to two special directions of the class of directions introduced earlier by Kojima, Shindoh, and Hara [13] using a different motivation. The second direction (with $P = S^{1/2}$), which is the KSH/HRVW/M-direction, was also proposed by Helmberg et al. [5] independently from [13]. To unify the above directions, including the one by Alizadeh, Haeberly, and Overton [2], Zhang [34] formally introduced the above scaling and symmetrization scheme for a general nonsingular scaling matrix P , which leads to a class of search directions parametrized by P . As suggested in Monteiro and Zhang [21], we refer to this set of search directions as the Monteiro–Zhang (MZ) family of search directions (or simply, the MZ-unified direction). In a recent paper, Todd, Toh, and Tütüncü [30] study conditions for the existence and uniqueness of search directions in the MZ-family and show that the NT-direction [26] is a member of this family corresponding to any scaling matrix P such that $P^T P = S^{1/2}(S^{1/2}XS^{1/2})^{-1/2}S^{1/2}$. More recently, Kojima, Shida, and Shindoh [11] show that the NT-direction also belongs to the class of directions introduced by Kojima, Shindoh, and Hara [13]. In contrast, it should be noted that the AHO-direction does not belong to the latter family.

This paper introduces new techniques for establishing the polynomiality of interior-point primal-dual SDP algorithms whose iterates belong to a narrow (or Frobenius norm) neighborhood of the central path. We illustrate our techniques by showing the polynomiality of two primal-dual feasible algorithms based on the MZ-unified direction: a short-step path-following method and a predictor-corrector method which are extensions of the LP algorithms studied in Kojima, Mizuno, and Yoshise [8], Monteiro and Adler [18, 19], and Mizuno, Todd, and Ye [16]. The iteration complexity bounds for both algorithms are shown to be exactly the same as their corresponding LP algorithms, namely, $\mathcal{O}(\sqrt{n}L)$ iterations to reduce the duality gap by a factor of at least $2^{-\mathcal{O}(L)}$. As opposed to Monteiro and Zhang [21], who establish an iteration-complexity bound for a long-step path-following method based on a subclass of the MZ-family of directions, namely, that corresponding to scaling matrices P such that $PXSP^{-1}$ is symmetric, our analysis is valid for the whole MZ-family of directions. Since this family includes the AHO-direction, we establish for the first time, as a special case of our unified analysis, the polynomial convergence of algorithms based on the AHO-direction. Finally, we mention that Kojima, Shida, and Shindoh [12] establish the global (but not polynomial) convergence of a primal-dual predictor-corrector

algorithm based on the AHO-direction whose iterates are restricted to a central path neighborhood narrower than the one used in this paper.

This paper is organized as follows. In section 2, we introduce the SDP problem, review the scaling and symmetrization scheme discussed above, and give some useful preliminary results. In section 3, we state and prove the technical results used in the polynomial convergence analysis of the algorithms of section 4. In section 4, we establish the polynomiality of two primal-dual feasible algorithms: the short-step path-following algorithm in subsection 4.1 and the predictor-corrector algorithm in subsection 4.2.

1.1. Notation and terminology. The following notation is used throughout the paper. The superscript T denotes transpose. $\mathbb{R}^p$ denotes the p -dimensional Euclidean space. The set of all $p \times q$ matrices with real entries is denoted by $\mathbb{R}^{p \times q}$. The set of all symmetric $p \times p$ matrices is denoted by $\mathcal{S}^p$. For $Q \in \mathcal{S}^p$, $Q \succeq 0$ means Q is positive semidefinite and $Q \succ 0$ means Q is positive definite. The trace of a matrix $Q \in \mathbb{R}^{p \times p}$ is denoted by $\text{Tr } Q \equiv \sum_{i=1}^n Q_{ii}$. For a matrix $Q \in \mathbb{R}^{p \times p}$ with all real eigenvalues, we denote its eigenvalues by $\lambda_i[Q]$, $i = 1, \dots, p$, and its smallest eigenvalue by $\lambda_{\min}[Q]$. Given P and Q in $\mathbb{R}^{p \times q}$, the inner product between them in the vector space $\mathbb{R}^{p \times q}$ is defined as $P \bullet Q \equiv \text{Tr } P^T Q$. The Euclidean norm and its associated operator norm are both denoted by $\|\cdot\|$; hence $\|Q\| \equiv \max_{\|u\|=1} \|Qu\|$ for any $Q \in \mathbb{R}^{p \times p}$. The Frobenius norm of $Q \in \mathbb{R}^{p \times p}$ is $\|Q\|_F \equiv (Q \bullet Q)^{1/2}$. $\mathcal{S}_+^p$ and $\mathcal{S}_{++}^p$ denote the set of all matrices in $\mathcal{S}^p$ which are positive semidefinite and positive definite, respectively. $\mathcal{S}_\perp^p$ denotes the set of all skew-symmetric matrices in $\mathbb{R}^{p \times p}$. Since $\mathcal{S}^p + \mathcal{S}_\perp^p = \mathbb{R}^{p \times p}$ and $U \bullet V = 0$ for every $U \in \mathcal{S}^p$ and $V \in \mathcal{S}_\perp^p$, it follows that $\mathcal{S}_\perp^p$ is the orthogonal complement of $\mathcal{S}^p$ with respect to the inner product $\bullet$.

2. The SDP problem and preliminary discussion. In this section, we describe the SDP problem considered in this paper and review the similar symmetrization operator introduced by Zhang [34] and the MZ-family of search directions.

This paper studies primal-dual path-following algorithms for solving the SDP problem

$$(2) \quad (\text{P}) \quad \min\{C \bullet X : A_i \bullet X = b_i, i = 1, \dots, m, X \succeq 0\},$$

and its associated dual SDP problem

$$(3) \quad (\text{D}) \quad \max \left\{ b^T y : \sum_{i=1}^m y_i A_i + S = C, S \succeq 0 \right\},$$

where $C \in \mathcal{S}^n$, $A_i \in \mathcal{S}^n$, $i = 1, \dots, m$, and $b = (b_1, \dots, b_m)^T \in \mathbb{R}^m$ are the data, and $X \in \mathcal{S}_+^n$ and $(S, y) \in \mathcal{S}_+^n \times \mathbb{R}^m$ are the primal and dual variables, respectively.

The set of *interior feasible solutions* of (2) and (3) are

$$F^0(\text{P}) \equiv \{X \in \mathcal{S} : A_i \bullet X = b_i, i = 1, \dots, m, X \succ 0\},$$

$$F^0(\text{D}) \equiv \left\{ (S, y) \in \mathcal{S} \times \mathbb{R}^m : \sum_{i=1}^m y_i A_i + S = C, S \succ 0 \right\},$$

respectively. Throughout this paper, we assume that $F^0(\text{P}) \times F^0(\text{D}) \neq \emptyset$ and that the matrices A_i , $i = 1, \dots, m$, are linearly independent. Under the first assumption, it is well known that both (2) and (3) have optimal solutions X^* and (y^*, S^*) such that $C \bullet X^* = b^T y^*$, i.e., the optimal values of (2) and (3) coincide. This last

condition, called the strong duality, can be alternatively expressed as $X^* \bullet S^* = 0$ or $X^* S^* = 0$. Hence, the set of primal and dual optimal solutions consist of all the solutions $(X, S, y) \in \mathcal{S}_+^n \times \mathcal{S}_+^n \times \mathbb{R}^m$ to the following optimality system:

$$(4a) \quad XS = 0,$$

$$(4b) \quad \sum_{i=1}^m y_i A_i + S - C = 0,$$

$$(4c) \quad A_i \bullet X - b_i = 0, \quad i = 1, \dots, m,$$

where (4a) is called the complementarity equation.

The left-hand side of the above system, viewed as a function of (X, S, y) , maps $\mathcal{S}^n \times \mathcal{S}^n \times \mathbb{R}^m$ into $\mathbb{R}^{n \times n} \times \mathcal{S}^n \times \mathbb{R}^m$, and hence it is a function between spaces of different dimensions. To apply Newton-type algorithms to solve (4), it is first necessary to symmetrize (4) so that the left-hand side of the resulting equivalent system becomes a function from $\mathcal{S}^n \times \mathcal{S}^n \times \mathbb{R}^m$ into itself.

Motivated by the works of Alizadeh, Haeberly, and Overton [2] and Monteiro [17], Zhang [34] introduced a general symmetrization scheme based on the so-called *similar symmetrization* operator $H_P : \mathbb{R}^{n \times n} \rightarrow \mathcal{S}^n$ defined as

$$H_P(M) \equiv \frac{1}{2} [PMP^{-1} + (PMP^{-1})^T] \quad \forall M \in \mathbb{R}^{n \times n},$$

where $P \in \mathbb{R}^{n \times n}$ is some nonsingular matrix. It has been shown by Zhang [34] that

$$H_P(M) = \tau I \iff M = \tau I$$

for any nonsingular matrix P , any matrix M with real spectrum (e.g., $M = XS$ with $X, S \in \mathcal{S}_+^n$), and any $\tau \in \mathbb{R}$. Consequently, for any given nonsingular matrix P , (4) is equivalent to the symmetric system

$$(5a) \quad H_P(XS) = 0,$$

$$(5b) \quad \sum_{i=1}^m y_i A_i + S - C = 0,$$

$$(5c) \quad A_i \bullet X - b_i = 0, \quad i = 1, \dots, m,$$

to which Newton-type methods can be applied. A perturbed Newton method applied to this system leads to the following linear system:

$$(6a) \quad H_P(\Delta XS + X\Delta S) = \sigma\mu I - H_P(XS),$$

$$(6b) \quad \sum_{i=1}^m \Delta y_i A_i + \Delta S = C - S - \sum_{i=1}^m y_i A_i,$$

$$(6c) \quad A_i \bullet \Delta X = b_i - A_i \bullet X, \quad i = 1, \dots, m,$$

where $(\Delta X, \Delta S, \Delta y) \in \mathcal{S}^n \times \mathcal{S}^n \times \mathbb{R}^m$, $\sigma \in [0, 1]$ is the centering parameter, and $\mu = \mu(X, S) \equiv (X \bullet S)/n$ is the normalized duality gap corresponding to (X, S, y) . Note that the solution $(\Delta X, \Delta S, \Delta y)$ of this linear system is the Newton direction at the point (X, S, y) with respect to a system of equations defining the (unique) point on the central path with duality gap $\sigma\mu$, namely, the system consisting of (5b), (5c) and $H_P(XS) = \sigma\mu I$.

The algorithms studied in this paper are all based on the centrality measure at a point $(X, S) \in \mathcal{S}_+^n \times \mathcal{S}_+^n$ defined as

$$(7) \quad d(X, S) \equiv \left\| X^{1/2} S X^{1/2} - \mu I \right\|_F = \left[\sum_{i=1}^n (\lambda_i[XS] - \mu)^2 \right]^{1/2}.$$

Given a constant $\gamma \in (0, 1)$, we let $\mathcal{N}_F(\gamma)$ denote the following narrow (or Frobenius) neighborhood of the central path:

$$\mathcal{N}_F(\gamma) \equiv \{(X, S, y) \in \mathcal{F}^0(P) \times \mathcal{F}^0(D) : d(X, S) \leq \gamma\mu\}.$$

The following simple result plays an important role in the polynomial convergence analysis of the algorithms of section 4.

LEMMA 2.1. *Suppose that $(X, S) \in \mathcal{S}_{++}^n \times \mathcal{S}_{++}^n$, $Q \in \mathbb{R}^{n \times n}$ is a nonsingular matrix, and $\mu \equiv (X \bullet S)/n$. Then*

- (a) $d(\tilde{X}, \tilde{S}) = d(X, S)$, where $\tilde{X} \equiv QXQ^T$ and $\tilde{S} \equiv Q^{-T}SQ^{-1}$;
- (b) $d(X, S) \leq \|H_Q(XS - \mu I)\|_F$ with equality holding if $QXSQ^{-1} \in \mathcal{S}^n$.

Proof. (a) Since $\tilde{X}\tilde{S} = QXSQ^{-1}$, that is, XS and $\tilde{X}\tilde{S}$ are similar matrices, we conclude that they have the same spectrum. This implies that $d(\tilde{X}, \tilde{S}) = d(X, S)$, since $d(X, S)$ can be expressed only in terms of the eigenvalues of XS .

(b) It has been shown in Lemma 3.3 of Monteiro [17] that $\|E\|_F \leq \|H_T(E)\|_F$ for any $E \in \mathcal{S}^n$ and any nonsingular $T \in \mathbb{R}^{n \times n}$. Using this inequality, with $E \equiv X^{1/2}SX^{1/2} - \mu I$ and $T \equiv QX^{1/2}$, and noting that $H_T(E) = H_Q(XS - \mu I)$, we obtain the inequality in (b). If $QXSQ^{-1} \in \mathcal{S}^n$, then $H_Q(XS - \mu I) = QXSQ^{-1} - \mu I$, and hence

$$\begin{aligned} \|H_Q(XS - \mu I)\|_F &= \|QXSQ^{-1} - \mu I\|_F = \left[\sum_{i=1}^n (\lambda_i[QXSQ^{-1}] - \mu)^2 \right]^{1/2} \\ &= \left[\sum_{i=1}^n (\lambda_i[XS] - \mu)^2 \right]^{1/2} = d(X, S). \quad \square \end{aligned}$$

3. Technical results. In this section we provide some technical results which will be used to establish the polynomial convergence of the algorithms presented in section 4.

We assume throughout this section that $(X, S, y) \in \mathcal{S}_{++}^n \times \mathcal{S}_{++}^n \times \mathbb{R}^m$, $P \in \mathbb{R}^{n \times n}$ is a nonsingular matrix, and $(\Delta X, \Delta S, \Delta y)$ is a solution of system (6) for some $\sigma \in [0, 1]$. Moreover, for $\alpha \in \mathbb{R}$, we let

$$(8) \quad \tilde{X} \equiv PXP^T, \quad \tilde{S} \equiv P^{-T}SP^{-1},$$

$$(9) \quad \widetilde{\Delta X} \equiv P\Delta XP^T, \quad \widetilde{\Delta S} \equiv P^{-T}\Delta SP^{-1},$$

$$(10) \quad X(\alpha) \equiv X + \alpha\Delta X, \quad S(\alpha) \equiv S + \alpha\Delta S, \quad y(\alpha) \equiv y + \alpha\Delta y,$$

$$(11) \quad \tilde{X}(\alpha) \equiv PX(\alpha)P^T = \tilde{X} + \alpha\widetilde{\Delta X},$$

$$(12) \quad \tilde{S}(\alpha) \equiv P^{-T}S(\alpha)P^{-1} = \tilde{S} + \alpha\widetilde{\Delta S},$$

$$(13) \quad W_X \equiv \tilde{X}^{-1/2} \left[\widetilde{\Delta X}\tilde{S} + \tilde{X}\widetilde{\Delta S} + \tilde{X}\tilde{S} - \sigma\mu I \right] \tilde{X}^{1/2},$$

$$(14) \quad \mu \equiv \frac{X \bullet S}{n} = \frac{\tilde{X} \bullet \tilde{S}}{n}, \quad \mu(\alpha) \equiv \frac{X(\alpha) \bullet S(\alpha)}{n} = \frac{\tilde{X}(\alpha) \bullet \tilde{S}(\alpha)}{n}.$$

LEMMA 3.1. $(\Delta X, \Delta S, \Delta y)$ satisfies system (6) if and only if $(\widetilde{\Delta X}, \widetilde{\Delta S}, \Delta y)$ is a solution of the system

$$(15a) \quad H_I(\widetilde{\Delta X} \widetilde{S} + \widetilde{X} \widetilde{\Delta S}) = \sigma \mu I - H_I(\widetilde{X} \widetilde{S}),$$

$$(15b) \quad \sum_{i=1}^m \Delta y_i \widetilde{A}_i + \widetilde{\Delta S} = \widetilde{C} - \widetilde{S} - \sum_{i=1}^m y_i \widetilde{A}_i,$$

$$(15c) \quad \widetilde{A}_i \bullet \widetilde{\Delta X} = b_i - \widetilde{A}_i \bullet \widetilde{X}, \quad i = 1, \dots, m,$$

where $\widetilde{C} \equiv P^{-T} C P^{-1}$ and $\widetilde{A}_i \equiv P^{-T} A_i P^{-1}$ for $i = 1, \dots, m$.

Proof. The proof follows immediately from (6), (8), and (9). $\square$

The result above says that $(\widetilde{\Delta X}, \widetilde{\Delta S}, \Delta y)$ is exactly the AHO-direction at the point $(\widetilde{X}, \widetilde{S}, y)$ for the pair of dual semidefinite programs with data $\widetilde{C}$ and $(\widetilde{A}_i, b_i)$, $i = 1, \dots, m$. As an immediate consequence of this lemma and Theorem 1 of Monteiro and Zanjácomo [20], we obtain the following result which ensures the existence and uniqueness of the solution of system (6) when (X, S, y) is well centered.

PROPOSITION 3.2. If $(X, S, y) \in \mathcal{S}_{++}^n \times \mathcal{S}_{++}^n \times \mathbb{R}^m$ is such that

$$\|S^{1/2} X S^{1/2} - \nu I\| \leq \frac{\nu}{2}$$

for some scalar $\nu > 0$, then system (6) has a unique solution.

Proof. It was shown in Theorem 1 of Monteiro and Zanjácomo [20] that system (15) has a unique solution whenever there exists $\nu > 0$ such that

$$\|\widetilde{S}^{1/2} \widetilde{X} \widetilde{S}^{1/2} - \nu I\| \leq \frac{\nu}{2}.$$

Since $\|S^{1/2} X S^{1/2} - \nu I\| = \|\widetilde{S}^{1/2} \widetilde{X} \widetilde{S}^{1/2} - \nu I\|$, the result follows from Lemma 3.1. $\square$

It should be observed that the above result can be strengthened if further conditions are imposed on the scaling matrix P . For example, it has been shown by Todd, Toh, and Tütüncü [30] (see also Monteiro and Zhang [21]) that system (6) has a unique solution for any $(X, S, y) \in \mathcal{S}_{++}^n \times \mathcal{S}_{++}^n \times \mathbb{R}^m$ and scaling matrix P such that $P X S P^{-1} \in \mathcal{S}^n$.

In the remaining part of this section, we impose the following condition (X, S, y) .

Assumption. $(X, S, y) \in \mathcal{F}^0(P) \times \mathcal{F}^0(D)$.

LEMMA 3.3. The following relations hold:

$$(16) \quad H_{\widetilde{X}^{1/2}}(W_X) = 0,$$

$$(17) \quad \widetilde{\Delta X} \bullet \widetilde{\Delta S} = \Delta X \bullet \Delta S = 0.$$

Proof. Relation (16) follows immediately from (15a). The two identities in (17) follow from relations (6b), (6c), and (9) and the assumption that $(X, S, y) \in \mathcal{F}^0(P) \times \mathcal{F}^0(D)$. $\square$

LEMMA 3.4. For every $\alpha \in \mathbb{R}$, we have

$$(18) \quad \mu(\alpha) = (1 - \alpha + \sigma \alpha) \mu,$$

$$(19) \quad \widetilde{X}^{-1/2} \left[\widetilde{X}(\alpha) \widetilde{S}(\alpha) - \mu(\alpha) I \right] \widetilde{X}^{1/2} = (1 - \alpha) \left(\widetilde{X}^{1/2} \widetilde{S} \widetilde{X}^{1/2} - \mu I \right) + \alpha W_X$$

$$(20) \quad + \alpha^2 \widetilde{X}^{-1/2} \widetilde{\Delta X} \widetilde{\Delta S} \widetilde{X}^{1/2}.$$

Proof. By (10), we have

$$X(\alpha)S(\alpha) = (X + \alpha\Delta X)(S + \alpha\Delta S) = XS + \alpha(X\Delta S + \Delta XS) + \alpha^2\Delta X\Delta S,$$

which, together with the linearity of $H_P(\cdot)$ and (6a), implies that

$$\begin{aligned} H_P(X(\alpha)S(\alpha)) &= H_P(XS) + \alpha H_P(X\Delta S + \Delta XS) + \alpha^2 H_P(\Delta X\Delta S) \\ &= H_P(XS) + \alpha H_P(\sigma\mu I - XS) + \alpha^2 H_P(\Delta X\Delta S) \\ &= (1 - \alpha)H_P(XS) + \alpha\sigma\mu I + \alpha^2 H_P(\Delta X\Delta S). \end{aligned}$$

Using this expression and the identity $\text{Tr } H_P(M) = \text{Tr } M$ for $M \in \Re^{n \times n}$, we obtain

$$\begin{aligned} X(\alpha) \bullet S(\alpha) &= \text{Tr } [X(\alpha)S(\alpha)] = \text{Tr } [H_P(X(\alpha)S(\alpha))] \\ &= \text{Tr } [(1 - \alpha)H_P(XS) + \alpha\sigma\mu I + \alpha^2 H_P(\Delta X\Delta S)] \\ &= (1 - \alpha)\text{Tr } H_P(XS) + \alpha\sigma\mu n + \alpha^2 \text{Tr } H_P(\Delta X\Delta S) \\ &= (1 - \alpha)X \bullet S + \alpha\sigma\mu n + \alpha^2 \Delta X \bullet \Delta S. \end{aligned}$$

Dividing this expression by n and noting (14) and (17), we obtain (18). Using (11), (12), (18), and (13), we conclude that, for every $\alpha \in \Re$,

$$\begin{aligned} \tilde{X}(\alpha)\tilde{S}(\alpha) - \mu(\alpha)I &= (\tilde{X} + \alpha\tilde{\Delta X})(\tilde{S} + \alpha\tilde{\Delta S}) - \mu(\alpha)I \\ &= \tilde{X}\tilde{S} + \alpha(\tilde{\Delta X}\tilde{S} + \tilde{X}\tilde{\Delta S}) + \alpha^2\tilde{\Delta X}\tilde{\Delta S} - (1 - \alpha + \sigma\alpha)\mu I \\ &= (1 - \alpha)(\tilde{X}\tilde{S} - \mu I) + \alpha(\tilde{\Delta X}\tilde{S} + \tilde{X}\tilde{\Delta S} + \tilde{X}\tilde{S} - \sigma\mu I) + \alpha^2\tilde{\Delta X}\tilde{\Delta S} \\ &= (1 - \alpha)(\tilde{X}\tilde{S} - \mu I) + \alpha\tilde{X}^{1/2}W_X\tilde{X}^{-1/2} + \alpha^2\tilde{\Delta X}\tilde{\Delta S}. \end{aligned}$$

Multiplying this identity on the left by $\tilde{X}^{-1/2}$ and on the right by $\tilde{X}^{1/2}$, we obtain (20). $\square$

The following technical result plays a major role in our analysis.

LEMMA 3.5. *Let $W \in \Re^{n \times n}$ be such that $H_Q(W) = 0$ for some nonsingular $Q \in \Re^{n \times n}$. Then,*

$$(21) \quad \|H_I(W)\|_F \leq \frac{1}{2} \|W - W^T\|_F,$$

$$(22) \quad \|W\|_F \leq \frac{\sqrt{2}}{2} \|W - W^T\|_F.$$

In particular, if $W = U_1 + U_2$ for some $U_1 \in \mathcal{S}^n$ and $U_2 \in \Re^{n \times n}$, then

$$\|W\|_F \leq \sqrt{2} \|U_2\|_F.$$

Proof. Define $U \equiv (W + W^T)/2 \in \mathcal{S}^n$ and $\hat{U} \equiv (W - W^T)/2 \in \mathcal{S}_\perp^n$. Clearly, $W = U + \hat{U}$ and $U \bullet \hat{U} = 0$. Using these relations, the assumption that $H_Q(W) = 0$, and the identity $\|B + B^T\|_F^2 = 2(\|B\|_F^2 + \text{Tr } B^2)$ for $B \in \Re^{n \times n}$, we obtain

$$\begin{aligned} 0 &= 2\|H_Q(W)\|_F^2 = \frac{1}{2} \|QWQ^{-1} + (QWQ^{-1})^T\|_F^2 \\ &= \|QWQ^{-1}\|_F^2 + \text{Tr } [(QWQ^{-1})^2] = \|QWQ^{-1}\|_F^2 + \text{Tr } [W^2] \\ &\geq \text{Tr } [(U + \hat{U})^2] = \text{Tr } [U^2 + U\hat{U} + \hat{U}U + \hat{U}^2] \\ &= \|U\|_F^2 + 2U \bullet \hat{U} + \text{Tr } \hat{U}^2 = \|U\|_F^2 - \|\hat{U}\|_F^2, \end{aligned}$$

and hence $\|U\|_F \leq \|\widehat{U}\|_F$; that is, (21) holds. Using this inequality and the fact that $U \bullet \widehat{U} = 0$, we obtain

$$\|W\|_F^2 = \|U + \widehat{U}\|_F^2 = \|U\|_F^2 + \|\widehat{U}\|_F^2 \leq 2\|\widehat{U}\|_F^2;$$

that is, (22) holds. To show the last part of the lemma, observe that $\|W - W^T\|_F = \|U_2 - U_2^T\|_F \leq 2\|U_2\|_F$, and hence, by (22), we obtain $\|W\|_F \leq \sqrt{2}\|U_2\|_F$. $\square$

Using the above result, we can now prove the following lemma.

LEMMA 3.6. *For every $\theta \in \mathfrak{R}$, we have*

$$(23) \quad \|W_X\|_F \leq \sqrt{2} \delta_x \left\| \widetilde{S}^{1/2} \widetilde{X}^{1/2} - \theta \mu \widetilde{S}^{-1/2} \widetilde{X}^{-1/2} \right\|$$

and

$$(24) \quad \begin{aligned} \left\| \widetilde{X}^{-1/2} [\widetilde{X}(\alpha) \widetilde{S}(\alpha) - \mu(\alpha) I] \widetilde{X}^{1/2} \right\|_F &\leq (1 - \alpha) d(\widetilde{X}, \widetilde{S}) + \alpha^2 \delta_x \delta_s \\ &+ \alpha \sqrt{2} \delta_x \left\| \widetilde{X}^{1/2} \widetilde{S}^{1/2} - \theta \mu \widetilde{X}^{-1/2} \widetilde{S}^{-1/2} \right\| \end{aligned}$$

for all $\alpha \in [0, 1]$, where

$$(25) \quad \delta_x \equiv \left\| \widetilde{X}^{-1/2} \widetilde{\Delta} \widetilde{X} \widetilde{S}^{1/2} \right\|_F, \quad \delta_s \equiv \left\| \widetilde{S}^{-1/2} \widetilde{\Delta} \widetilde{S} \widetilde{X}^{1/2} \right\|_F.$$

Proof. Let $\theta \in \mathfrak{R}$ be given. By (13), we have $W_X = U_1 + U_2$, where

$$\begin{aligned} U_1 &\equiv \widetilde{X}^{1/2} \widetilde{\Delta} \widetilde{S} \widetilde{X}^{1/2} + \theta \mu \widetilde{X}^{-1/2} \widetilde{\Delta} \widetilde{X} \widetilde{X}^{-1/2} + \widetilde{X}^{1/2} \widetilde{S} \widetilde{X}^{1/2} - \sigma \mu I \in \mathcal{S}^n, \\ U_2 &\equiv \widetilde{X}^{-1/2} \widetilde{\Delta} \widetilde{X} \widetilde{S}^{1/2} \left(\widetilde{S}^{1/2} \widetilde{X}^{1/2} - \theta \mu \widetilde{S}^{-1/2} \widetilde{X}^{-1/2} \right). \end{aligned}$$

By (16), we have $H_Q(W_X) = 0$ for $Q \equiv \widetilde{X}^{1/2}$. These observations, together with Lemma 3.5, the definition of U_2 and δ_x , and the fact that $\|AB\|_F \leq \|A\|_F \|B\|_F$ for every $A, B \in \mathfrak{R}^{n \times n}$ (see exercise 20 of section 5.6 of [6]), imply that

$$\|W_X\|_F \leq \sqrt{2} \|U_2\|_F \leq \sqrt{2} \delta_x \left\| \widetilde{S}^{1/2} \widetilde{X}^{1/2} - \theta \mu \widetilde{S}^{-1/2} \widetilde{X}^{-1/2} \right\|;$$

i.e., (23) holds. Moreover, relations (7) and (20), the fact that $\alpha \in [0, 1]$, and the definitions of δ_x and δ_s , yield

$$\begin{aligned} &\left\| \widetilde{X}^{-1/2} [\widetilde{X}(\alpha) \widetilde{S}(\alpha) - \mu(\alpha) I] \widetilde{X}^{1/2} \right\|_F \\ &\leq (1 - \alpha) \left\| \widetilde{X}^{1/2} \widetilde{S} \widetilde{X}^{1/2} - \mu I \right\|_F + \alpha \|W_X\|_F + \alpha^2 \delta_x \delta_s \\ &= (1 - \alpha) d(\widetilde{X}, \widetilde{S}) + \alpha \|W_X\|_F + \alpha^2 \delta_x \delta_s, \end{aligned}$$

which, together with (23), imply (24). $\square$

To shorten the notation used in the results below, we let

$$(26) \quad \Phi_\theta(A, B) \equiv \left\| A^{1/2} B^{1/2} - \theta \frac{A \bullet B}{n} A^{-1/2} B^{-1/2} \right\|_F,$$

for every $(A, B) \in \mathcal{S}_{++}^n \times \mathcal{S}_{++}^n$ and $\theta \in \Re$.

LEMMA 3.7. *If $d(X, S) \leq \gamma\mu$ for some $\gamma \in (0, 1)$, then*

$$(27) \quad \left\| \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\|^2 \leq \frac{1}{(1-\gamma)\mu},$$

$$(28) \quad \left[\Phi_\theta(\tilde{X}, \tilde{S}) \right]^2 \leq \frac{\gamma^2 + (1-\theta)^2 n}{1-\gamma} \mu.$$

Proof. By relation (7) and the assumption that $d(X, S) \leq \gamma\mu$, we have $\lambda_{\min}[XS] \geq (1-\gamma)\mu$. This, together with (8), implies

$$\begin{aligned} \left\| \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\|^2 &= \left\| \tilde{X}^{-1/2} \tilde{S}^{-1} \tilde{X}^{-1/2} \right\| = \frac{1}{\lambda_{\min}[\tilde{X}^{1/2} \tilde{S} \tilde{X}^{1/2}]} = \frac{1}{\lambda_{\min}[\tilde{X} \tilde{S}]} \\ &= \frac{1}{\lambda_{\min}[PXS P^{-1}]} = \frac{1}{\lambda_{\min}[XS]} \leq \frac{1}{(1-\gamma)\mu}; \end{aligned}$$

that is, (27) holds. Using relations (26), (14), (27), (7), the fact that the matrices $\tilde{X}^{1/2} \tilde{S} \tilde{X}^{1/2} - \mu I$ and I are orthogonal, the assumption that $d(X, S) \leq \gamma\mu$, and Lemma 2.1(a), we obtain

$$\begin{aligned} \left[\Phi_\theta(\tilde{X}, \tilde{S}) \right]^2 &= \left\| \tilde{X}^{1/2} \tilde{S}^{1/2} - \theta \mu \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\|_F^2 \\ &\leq \left\| \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\|^2 \left\| \tilde{X}^{1/2} \tilde{S} \tilde{X}^{1/2} - \theta \mu I \right\|_F^2 \\ &\leq \frac{1}{(1-\gamma)\mu} \left(\left\| \tilde{X}^{1/2} \tilde{S} \tilde{X}^{1/2} - \mu I \right\|_F^2 + \left\| \mu I - \theta \mu I \right\|_F^2 \right) \\ &= \frac{1}{(1-\gamma)\mu} (d(X, S)^2 + (1-\theta)^2 \mu^2 n) \leq \frac{\gamma^2 + (1-\theta)^2 n}{1-\gamma} \mu; \end{aligned}$$

that is, (28) holds. $\square$

LEMMA 3.8. *If $(X, S, y) \in \mathcal{N}_F(\gamma)$ for some $\gamma > 0$ satisfying*

$$(29) \quad 2\sqrt{2} \frac{\gamma}{1-\gamma} \leq 1,$$

then

$$\max\{\delta_x, \delta_s\} \leq 2\Phi_\sigma(\tilde{X}, \tilde{S}),$$

where δ_x , δ_s , and $\Phi_\sigma(\cdot, \cdot)$ are defined in (25) and (26).

Proof. Using (13), it is easy to see that

$$(30) \quad \tilde{X}^{1/2} \widetilde{\Delta S} \tilde{S}^{-1/2} + \tilde{X}^{-1/2} \widetilde{\Delta X} \tilde{S}^{1/2} = W_X \tilde{X}^{-1/2} \tilde{S}^{-1/2} + \sigma \mu \tilde{X}^{-1/2} \tilde{S}^{-1/2} - \tilde{X}^{1/2} \tilde{S}^{1/2}.$$

By (17), it is easy to see that the two terms on the left-hand side of (30) are orthogonal. This observation, together with (25), (30), (26), (14), (23) with $\theta = 1$, (27), (28) with $\theta = 1$, and (29), implies that

$$\begin{aligned}
\max\{\delta_x, \delta_s\} &\leq (\delta_x^2 + \delta_s^2)^{1/2} = \left(\left\| \tilde{X}^{-1/2} \widetilde{\Delta X} \tilde{S}^{1/2} \right\|_F^2 + \left\| \tilde{S}^{-1/2} \widetilde{\Delta S} \tilde{X}^{1/2} \right\|_F^2 \right)^{1/2} \\
&= \left\| \tilde{X}^{-1/2} \widetilde{\Delta X} \tilde{S}^{1/2} + \tilde{X}^{1/2} \widetilde{\Delta S} \tilde{S}^{-1/2} \right\|_F \\
&\leq \|W_X\|_F \left\| \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\| + \left\| \tilde{X}^{1/2} \tilde{S}^{1/2} - \sigma \mu \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\|_F \\
&\leq \sqrt{2} \delta_x \left\| \tilde{X}^{1/2} \tilde{S}^{1/2} - \mu \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\|_F \left\| \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\| + \Phi_\sigma(\tilde{X}, \tilde{S}) \\
&\leq \sqrt{2} \delta_x \frac{\gamma \mu^{1/2}}{(1-\gamma)^{1/2}} \frac{1}{(1-\gamma)^{1/2} \mu^{1/2}} + \Phi_\sigma(\tilde{X}, \tilde{S}) \\
&\leq \frac{\delta_x}{2} + \Phi_\sigma(\tilde{X}, \tilde{S}) \leq \frac{1}{2} \max\{\delta_x, \delta_s\} + \Phi_\sigma(\tilde{X}, \tilde{S}),
\end{aligned}$$

from which the lemma follows. $\square$

As an immediate consequence of the above lemmas, we obtain the following result.

LEMMA 3.9. *If $(X, S, y) \in \mathcal{N}_F(\gamma)$ for some $\gamma > 0$ satisfying (29), then, for every $\alpha \in [0, 1]$, we have*

$$\begin{aligned}
&\left\| \tilde{X}^{-1/2} [\tilde{X}(\alpha) \tilde{S}(\alpha) - \mu(\alpha)] \tilde{X}^{1/2} \right\|_F \\
&\leq \left\{ (1-\alpha)\gamma + 2\sqrt{2}\alpha \frac{\gamma [\gamma^2 + (1-\sigma)^2 n]^{1/2}}{1-\gamma} + 4\alpha^2 \frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma} \right\} \mu.
\end{aligned}$$

Proof. Using (24) with $\theta = 1$, Lemma 2.1(a), (8), the assumption that $d(X, S) \leq \gamma\mu$, Lemma 3.8, and (28) with $\theta = 1$ and $\theta = \sigma$, we obtain

$$\begin{aligned}
&\left\| \tilde{X}^{-1/2} [\tilde{X}(\alpha) \tilde{S}(\alpha) - \mu(\alpha)] \tilde{X}^{1/2} \right\|_F \\
&\leq (1-\alpha) d(X, S) + \sqrt{2}\alpha \delta_x \Phi_1(\tilde{X}, \tilde{S}) + \alpha^2 \delta_x \delta_s \\
&\leq (1-\alpha)\gamma\mu + 2\sqrt{2}\alpha \Phi_\sigma(\tilde{X}, \tilde{S}) \Phi_1(\tilde{X}, \tilde{S}) + 4\alpha^2 \left[\Phi_\sigma(\tilde{X}, \tilde{S}) \right]^2 \\
&\leq \left\{ (1-\alpha)\gamma + 2\sqrt{2}\alpha \frac{\gamma [\gamma^2 + (1-\sigma)^2 n]^{1/2}}{1-\gamma} + 4\alpha^2 \frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma} \right\} \mu. \quad \square
\end{aligned}$$

4. Algorithms and polynomial convergence. In this section, we establish polynomial iteration-complexity bounds for two primal-dual feasible interior-point algorithms based on the MZ-family of search directions given by (6). Both algorithms are extensions of well-known algorithms for linear programming: the first one is a short-step path-following method which generalizes the algorithms presented in Kojima, Mizuno, and Yoshise [8] and Monteiro and Adler [18, 19]; the second one is a predictor-corrector algorithm similar to the predictor-corrector LP method of Mizuno, Todd, and Ye [16].

4.1. Short-step path-following algorithm. In this subsection, we analyze the polynomial convergence of a short-step path-following algorithm based on the MZ-unified search direction (6).

We start by stating the algorithm that will be considered in this subsection.

ALGORITHM I.

Choose constants γ and δ in $(0, 1)$ satisfying the conditions of Corollary 4.2 below and let $\sigma \equiv 1 - \delta/\sqrt{n}$. Let $L > 1$ and $(X^0, S^0, y^0) \in \mathcal{F}^0(P) \times \mathcal{F}^0(D)$ be such that $d(X^0, S^0) \leq \gamma\mu_0$, where $\mu_0 \equiv (X^0 \bullet S^0)/n$.

Repeat until $\mu_k \leq 2^{-L}\mu_0$, **do**

- (1) Choose a nonsingular matrix $P^k \in \mathbb{R}^{n \times n}$;
- (2) Compute the solution $(\Delta X^k, \Delta S^k, \Delta y^k)$ of system (6) with $P = P^k$, $\mu = \mu_k$, and $(X, S, y) = (X^k, S^k, y^k)$;
- (3) Set $(X^{k+1}, y^{k+1}, S^{k+1}) \equiv (X^k, y^k, S^k) + (\Delta X^k, \Delta S^k, \Delta y^k)$;
- (4) Set $\mu_{k+1} \equiv (X^{k+1} \bullet S^{k+1})/n$ and increment k by 1.

End

Setting $\Gamma = \gamma$ in the following result, we obtain the analysis of one iteration of Algorithm I for suitable choices of the constants γ and δ .

THEOREM 4.1. *Let $\gamma, \Gamma \in (0, 1)$ and $\delta \in [0, n^{1/2})$ be constants satisfying*

$$(31) \quad \frac{7(\gamma^2 + \delta^2)}{1 - \gamma} \leq \left(1 - \frac{\delta}{\sqrt{n}}\right) \Gamma, \quad \frac{2\sqrt{2}\gamma}{1 - \gamma} \leq 1.$$

Suppose that $(X, S, y) \in \mathcal{N}_F(\gamma)$, and let $(\Delta X, \Delta S, \Delta y)$ denote the solution of system (6), where $\sigma \equiv 1 - \delta/\sqrt{n}$, $\mu \equiv (X \bullet S)/n$, and $P \in \mathbb{R}^{n \times n}$ is an invertible matrix. Then,

- (a) $(\hat{X}, \hat{S}, \hat{y}) \equiv (X + \Delta X, S + \Delta S, y + \Delta y) \in \mathcal{N}_F(\Gamma)$;
- (b) $\hat{X} \bullet \hat{S} = (1 - \delta/\sqrt{n})(X \bullet S)$.

Proof. It follows from Lemma 3.9, the definition of σ , and (31) that for every $\alpha \in [0, 1]$,

$$(32) \quad \begin{aligned} \left\| \tilde{X}^{-1/2} [\tilde{X}(\alpha) \tilde{S}(\alpha) - \mu(\alpha)] \tilde{X}^{1/2} \right\|_F &\leq \left\{ (1 - \alpha)\gamma + 7\alpha \frac{\gamma^2 + (1 - \sigma)^2 n}{1 - \gamma} \right\} \mu \\ &= \left\{ (1 - \alpha)\gamma + \alpha \frac{7(\gamma^2 + \delta^2)}{1 - \gamma} \right\} \mu \\ &\leq \left\{ (1 - \alpha)\gamma + \alpha \left(1 - \frac{\delta}{\sqrt{n}}\right) \Gamma \right\} \mu \\ &= \{(1 - \alpha)\gamma + \alpha\Gamma\sigma\} \mu, \end{aligned}$$

and hence, in view of (18),

$$\left\| \tilde{X}^{-1/2} [\tilde{X}(\alpha) \tilde{S}(\alpha) - \mu(\alpha)] \tilde{X}^{1/2} \right\|_F \leq \max\{\gamma, \Gamma\} (1 - \alpha + \alpha\sigma) \mu = \max\{\gamma, \Gamma\} \mu(\alpha).$$

This relation implies that the matrix

$$G(\alpha) \equiv \left(\tilde{X}^{-1/2} \tilde{X}(\alpha) \tilde{S}(\alpha) \tilde{X}^{1/2} \right) / \mu(\alpha)$$

satisfies $\|G(\alpha) - I\| \leq \max\{\gamma, \Gamma\} < 1$, and thus that $G(\alpha)$ is nonsingular for every $\alpha \in [0, 1]$. Hence, we conclude that $\tilde{X}(\alpha)$ and $\tilde{S}(\alpha)$ are also nonsingular for every $\alpha \in [0, 1]$. A trivial continuity argument together with (11) and (12) shows that $(\tilde{X}(\alpha), \tilde{S}(\alpha))$ and $(X(\alpha), S(\alpha))$ are in $\mathcal{S}_{++}^n \times \mathcal{S}_{++}^n$, for every $\alpha \in [0, 1]$. Moreover, using (6b) and (6c) and the fact that $(X, S, y) \in \mathcal{F}^0(P) \times \mathcal{F}^0(D)$, we easily see that $(X(\alpha), S(\alpha), y(\alpha))$ satisfies (5b) and (5c), and hence that $(X(\alpha), S(\alpha), y(\alpha)) \in \mathcal{F}^0(P) \times \mathcal{F}^0(D)$, for every $\alpha \in [0, 1]$. Using Lemma 2.1(a) with $(X, S) = (X(1), S(1))$ and $Q = P$, Lemma 2.1(b)

with $(X, S) = (\tilde{X}(1), \tilde{S}(1))$ and $Q = \tilde{X}^{-1/2}$, and relations (11), (12), (18) with $\alpha = 1$, and (32) with $\alpha = 1$, we obtain

$$\begin{aligned} d(X(1), S(1)) = d(\tilde{X}(1), \tilde{S}(1)) &\leq \left\| H_{\tilde{X}^{-1/2}}[\tilde{X}(1)\tilde{S}(1) - \mu(1)I] \right\|_F \\ &\leq \left\| \tilde{X}^{-1/2}[\tilde{X}(1)\tilde{S}(1) - \mu(1)]\tilde{X}^{1/2} \right\|_F \leq \Gamma\mu(1). \end{aligned}$$

We have thus shown that $(\hat{X}, \hat{S}, \hat{y}) = (X(1), S(1), y(1)) \in \mathcal{N}_F(\Gamma)$; that is, (a) holds. Statement (b) is an immediate consequence of (18) with $\alpha = 1$ and the definition of σ . $\square$

As an immediate consequence of Theorem 4.1, we have the following convergence result for Algorithm I.

COROLLARY 4.2. *Assume that γ and δ are constants in $(0, 1)$ satisfying*

$$\frac{7(\gamma^2 + \delta^2)}{1 - \gamma} \leq \left(1 - \frac{\delta}{\sqrt{n}}\right)\gamma, \quad \frac{2\sqrt{2}\gamma}{1 - \gamma} \leq 1.$$

Then, every iterate (X^k, S^k, y^k) generated by Algorithm I is in the neighborhood $\mathcal{N}_F(\gamma)$ and satisfies $X^k \bullet S^k = (1 - \delta/\sqrt{n})^k (X^0 \bullet S^0)$. Moreover, Algorithm I terminates in at most $\mathcal{O}(\sqrt{n}L)$ iterations.

Examples of constants γ and δ satisfying the conditions of Corollary 4.2 are $\gamma = \delta = 1/20$.

4.2. Predictor-corrector algorithm. In this subsection, we give the polynomial convergence analysis of a predictor-corrector algorithm which is a direct extension of the LP predictor-corrector algorithm studied by Mizuno, Todd, and Ye [16].

The algorithm considered in this subsection is as follows.

ALGORITHM II.

Choose a constant $0 < \tau < 1$ satisfying the conditions of Theorem 4.3 below. Let $L > 1$ and $(X^0, S^0, y^0) \in \mathcal{F}^0(P) \times \mathcal{F}^0(D)$ be such that $d(X^0, S^0) \leq \tau\mu_0$, where $\mu_0 \equiv (X^0 \bullet S^0)/n$.

Repeat until $\mu_k \leq 2^{-L}\mu_0$, do

- (1) Choose a nonsingular matrix $P^k \in \mathbb{R}^{n \times n}$;
- (2) Compute the solution $(\Delta X^k, \Delta S^k, \Delta y^k)$ of system (6) with $P = P^k$, $\sigma = 0$, and $(X, S, y) = (X^k, S^k, y^k)$;
- (3) Let $\alpha_k \equiv \max\{\alpha \in [0, 1] : (X^k(\alpha'), S^k(\alpha'), y^k(\alpha')) \in \mathcal{N}_F(2\tau), \forall \alpha' \in [0, \alpha]\}$, where $(X^k(\alpha), S^k(\alpha), y^k(\alpha)) \equiv (X^k + \alpha\Delta X^k, S^k + \alpha\Delta S^k, y^k + \alpha\Delta y^k)$;
- (4) Let $(\hat{X}^k, \hat{S}^k, \hat{y}^k) \equiv (X^k, S^k, y^k) + \alpha_k(\Delta X^k, \Delta S^k, \Delta y^k)$;
- (5) Choose a nonsingular matrix $\hat{P}^k \in \mathbb{R}^{n \times n}$;
- (6) Compute the solution $(\widehat{\Delta X}^k, \widehat{\Delta S}^k, \widehat{\Delta y}^k)$ of system (6) with $P = \hat{P}^k$, $\sigma = 1$, and $(X, S, y) = (\hat{X}^k, \hat{S}^k, \hat{y}^k)$;
- (7) Set $(X^{k+1}, S^{k+1}, y^{k+1}) \equiv (\hat{X}^k, \hat{S}^k, \hat{y}^k) + (\widehat{\Delta X}^k, \widehat{\Delta S}^k, \widehat{\Delta y}^k)$;
- (8) Set $\mu_{k+1} \equiv (X^{k+1} \bullet S^{k+1})/n$ and increment k by 1.

End

The following result provides the polynomial convergence analysis of the above algorithm.

THEOREM 4.3. *Assume that $\tau \in (0, 1/30]$. Then, Algorithm II satisfies the following statements:*

- (a) for every $k \geq 0$, $(X^k, S^k, y^k) \in \mathcal{N}_F(\tau)$ and $(\widehat{X}^k, \widehat{S}^k, \widehat{y}^k) \in \mathcal{N}_F(2\tau)$;
 (b) for every $k \geq 0$, $X^{k+1} \bullet S^{k+1} = \widehat{X}^k \bullet \widehat{S}^k = (1 - \alpha_k)(X^k \bullet S^k)$ and

$$\alpha_k = \frac{1}{\mathcal{O}(\sqrt{n})};$$

- (c) the algorithm terminates in at most $\mathcal{O}(\sqrt{n}L)$ iterations.

Proof. Statement (c) and the well-definedness of Algorithm II follow directly from (a) and (b). In turn, these two statements follow by a simple induction argument and the two lemmas below. $\square$

The following lemma analyzes the predictor step of Algorithm II, namely, the step described in items (1)–(4) of Algorithm II.

LEMMA 4.4. Suppose that $(X, S, y) \in \mathcal{N}_F(\tau)$ for some $\tau \in (0, 1/2)$. Let the triple $(\Delta X, \Delta S, \Delta y)$ denote the solution of (6) with $\sigma = 0$ and for some invertible matrix $P \in \mathbb{R}^{n \times n}$. Let $\bar{\alpha}$ denote the unique positive root of the second-order polynomial $p(\alpha)$ defined as

$$(33) \quad p(\alpha) \equiv 4 \frac{(\tau^2 + n)}{1 - \tau} \alpha^2 + \left(2\sqrt{2}\tau \frac{(\tau^2 + n)^{1/2}}{1 - \tau} + \tau \right) \alpha - \tau.$$

Then, for any $\alpha \in [0, \bar{\alpha}]$, we have

- (a) $(X(\alpha), S(\alpha), y(\alpha)) \in \mathcal{N}_F(2\tau)$;
 (b) $X(\alpha) \bullet S(\alpha) = (1 - \alpha)(X \bullet S)$.

Moreover, $\bar{\alpha} = 1/\mathcal{O}(n^{1/2})$.

Proof. Using Lemma 3.9 with $\gamma = \tau$ and $\sigma = 0$, the fact that $p(\alpha) \leq 0$ for $\alpha \in [0, \bar{\alpha}]$, and relation (18) with $\sigma = 0$ and (33), we obtain

$$\begin{aligned} & \left\| \tilde{X}^{-1/2} [\tilde{X}(\alpha) \tilde{S}(\alpha) - \mu(\alpha)] \tilde{X}^{1/2} \right\|_F \\ & \leq \left\{ (1 - \alpha)\tau + 2\sqrt{2}\alpha \frac{\tau(\tau^2 + n)^{1/2}}{1 - \tau} + 4\alpha^2 \frac{\tau^2 + n}{1 - \tau} \right\} \mu \\ & = 2\tau\mu(\alpha) + p(\alpha)\mu \leq 2\tau\mu(\alpha). \end{aligned}$$

An argument similar to the one used in Theorem 4.1, together with the fact that $2\tau < 1$, can be used to show that (a) holds. Statement (b) follows from (18) with $\sigma = 0$. The last assertion of the lemma follows by a straightforward verification. $\square$

The following lemma analyzes the corrector step of Algorithm II, namely, the step described in items (5)–(7) of Algorithm II.

LEMMA 4.5. Suppose $(\widehat{X}, \widehat{S}, \widehat{y})$ is in $\mathcal{N}_F(2\tau)$ for some $\tau \in (0, 1/30]$. Let the triple $(\widehat{\Delta X}, \widehat{\Delta S}, \widehat{\Delta y})$ denote the solution of (6) with $(X, S, y) = (\widehat{X}, \widehat{S}, \widehat{y})$, $\sigma = 1$, and $P = \widehat{P}$, for some nonsingular matrix $\widehat{P} \in \mathbb{R}^{n \times n}$. Then,

$$\begin{aligned} & (\widehat{X}, \widehat{S}, \widehat{y}) + (\widehat{\Delta X}, \widehat{\Delta S}, \widehat{\Delta y}) \in \mathcal{N}_F(\tau), \\ & (\widehat{X} + \widehat{\Delta X}) \bullet (\widehat{S} + \widehat{\Delta S}) = \widehat{X} \bullet \widehat{S}. \end{aligned}$$

Proof. The result follows immediately from Theorem 4.1 with $\delta = 0$, $(X, S, y) = (\widehat{X}, \widehat{S}, \widehat{y})$, $\gamma = 2\tau$ and $\Gamma = \tau$, and the fact that (31) holds when $0 < \tau \leq 1/30$. $\square$

5. Remarks. To keep the presentation simple and the analysis more transparent, we have focused our attention only on feasible methods. It would be interesting to check whether the techniques developed in this paper could also be used to establish the polynomial convergence of infeasible primal-dual path-following algorithms based on the MZ-unified direction.

The analysis of this paper makes strong use of the fact that the algorithms are based on the narrow neighborhood of the central path. The question of whether long-step path-following methods based on the MZ-unified direction can be proved to be “polynomially convergent” under weaker conditions than those imposed in Monteiro and Zhang [21] is an interesting topic for future research.

We end this section by stating a technical result whose proof is similar to that of Lemma 3.5 and which may eventually have application elsewhere. To state this result, define for every nonsingular matrix $P \in \mathbb{R}^{n \times n}$ the operator $A_P : \mathbb{R}^{n \times n} \rightarrow \mathcal{S}_\perp^n$ as

$$A_P(M) \equiv \frac{1}{2} [PMP^{-1} - (PMP^{-1})^T] \quad \forall M \in \mathbb{R}^{n \times n}.$$

LEMMA 5.1. *Let $W \in \mathbb{R}^{n \times n}$ be such that $A_Q(W) = 0$ for some nonsingular $Q \in \mathbb{R}^{n \times n}$. Then,*

$$\begin{aligned} \|A_I(W)\|_F &\leq \|H_I(W)\|_F, \\ \|W\|_F &\leq \sqrt{2} \|H_I(W)\|_F. \end{aligned}$$

In particular, if $W = V_1 + V_2$ for some $V_1 \in \mathcal{S}_\perp^n$ and $V_2 \in \mathbb{R}^{n \times n}$, then

$$\|W\|_F \leq \sqrt{2} \|V_2\|_F.$$

Proof. The proof is the same as the proof of Lemma 3.5 but instead of using the identity $\|B + B^T\|_F^2 = 2(\|B\|_F^2 + \text{Tr } B^2)$ uses the alternative identity $\|B - B^T\|_F^2 = 2(\|B\|_F^2 - \text{Tr } B^2)$. $\square$

Note that, in terms of the operator A_P , the inequalities (21) and (22) can be simply written as

$$\begin{aligned} \|H_I(W)\|_F &\leq \|A_I(W)\|_F, \\ \|W\|_F &\leq \sqrt{2} \|A_I(W)\|_F, \end{aligned}$$

respectively.

Acknowledgments. This work was done while the author was visiting the Institute of Statistical Mathematics and the Tokyo Institute of Technology in Japan. He is grateful to both institutes for providing a congenial scientific atmosphere during his stay. He would like to thank his host Dr. Takashi Tsuchiya for making his visit to Japan a very enjoyable experience and for having many useful discussions which have been invaluable toward the development of this work. Finally, thanks also are due to Prof. Masakazu Kojima for making possible the author’s living at the International House in the Tokyo Institute of Technology. Finally, the financial support of the National Science Foundation and Office of Naval Research is gratefully acknowledged.

REFERENCES

- [1] F. ALIZADEH, *Interior point methods in semidefinite programming with applications to combinatorial optimization*, SIAM J. Optim., 5 (1995), pp. 13–51.
- [2] F. ALIZADEH, J.-P. HAEBERLY, AND M. OVERTON, *Primal-dual Interior-point Methods for Semidefinite Programming*, Technical Report 659, Computer Science Department, Courant Institute of Mathematical Sciences, New York University, New York, NY, 1994.
- [3] F. ALIZADEH, J.-P. HAEBERLY, AND M. OVERTON, *Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results*, SIAM J. Optim., 8 (1998), pp. 746–768.
- [4] R. M. FREUND, *Complexity of an Algorithm for Finding an Approximate Solution of a Semidefinite Program with No Regularity Condition*, Working paper OR 302-94, Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, December 1994.
- [5] C. HELMBERG, F. RENDL, R. J. VANDERBEI, AND H. WOLKOWICZ, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.
- [6] R. A. HORN AND C. R. JOHNSON, *Matrix Analysis*, Cambridge University Press, New York, 1985.
- [7] F. JARRE, *An interior-point method for minimizing the maximum eigenvalue of a linear combination of matrices*, SIAM J. Control Optim., 31 (1993), pp. 1360–1377.
- [8] M. KOJIMA, S. MIZUNO, AND A. YOSHISE, *A polynomial-time algorithm for a class of linear complementarity problems*, Math. Programming, 44 (1989), pp. 1–26.
- [9] M. KOJIMA, S. MIZUNO, AND A. YOSHISE, *A primal-dual interior point algorithm for linear programming*, in *Progress in Mathematical Programming: Interior Point and Related Methods*, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 29–47.
- [10] M. KOJIMA, M. SHIDA, AND S. SHINDOH, *Local convergence of predictor-corrector infeasible-interior-point algorithms for SDPs and SDLCPs*, Math. Programming, 80 (1998), pp. 129–160.
- [11] M. KOJIMA, M. SHIDA, AND S. SHINDOH, *A Note on the Nesterov-Todd and the Kojima-Shindoh-Hara Search Directions in Semidefinite Programming*, Research Report B-313, Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, Oh-okayama, Meguro-ku, Tokyo, Japan, April 1996.
- [12] M. KOJIMA, M. SHIDA, AND S. SHINDOH, *A predictor-corrector interior-point algorithm for the semidefinite linear complementarity problem using the Alizadeh-Haeberly-Overton search direction*, SIAM J. Optim., to appear.
- [13] M. KOJIMA, S. SHINDOH, AND S. HARA, *Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.
- [14] C.-J. LIN AND R. SAIGAL, *A Predictor-corrector Method for Semi-definite Programming*, Working paper, Dept. of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, MI, 1995.
- [15] Z.-Q. LUO, J. F. STURM, AND S. ZHANG, *Superlinear convergence of a symmetric primal-dual path following algorithm for semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 59–81.
- [16] S. MIZUNO, M. J. TODD, AND Y. YE, *On adaptive step primal-dual interior-point algorithms for linear programming*, Math. Oper. Res., 18 (1993), pp. 945–981.
- [17] R. D. C. MONTEIRO, *Primal-dual path following algorithms for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 663–678.
- [18] R. D. C. MONTEIRO AND I. ADLER, *Interior path-following primal-dual algorithms. Part I: Linear programming*, Math. Programming, 44 (1989), pp. 27–41.
- [19] R. D. C. MONTEIRO AND I. ADLER, *Interior path-following primal-dual algorithms. Part II: Convex quadratic programming*, Math. Programming, 44 (1989), pp. 43–66.
- [20] R. D. C. MONTEIRO AND P. ZANJÁCOMO, *A note on the existence of the Alizadeh-Haeberly-Overton direction for semidefinite programming*, Math. Programming, 78 (1997), pp. 393–396.
- [21] R. D. C. MONTEIRO AND Y. ZHANG, *A unified analysis for a class of path-following primal-dual interior-point algorithms for semidefinite programming*, Math. Programming, 81 (1998), pp. 281–299.
- [22] Y. E. NESTEROV AND A. S. NEMIROVSKII, *Polynomial barrier methods in convex programming*, Ekonomika i Matem. Metody, 24 (1988), pp. 1084–1091 (in Russian; English translation in Matekon: Translations of Russian and East European Math. Economics).
- [23] Y. E. NESTEROV AND A. S. NEMIROVSKII, *Self-concordant Functions and Polynomial Time Methods in Convex Programming*, preprint, Central Economic & Mathematical Institute, USSR Acad. Sci. Moscow, USSR, 1989.

- [24] Y. E. NESTEROV AND A. S. NEMIROVSKII, *Optimization Over Positive Semidefinite Matrices: Mathematical Background and User's Manual*, Technical report, Central Economic & Mathematical Institute, USSR Acad. Sci. Moscow, USSR, 1990.
- [25] Y. E. NESTEROV AND A. S. NEMIROVSKII, *Interior Point Methods in Convex Programming: Theory and Applications*, SIAM, Philadelphia, PA, 1994.
- [26] Y. E. NESTEROV AND M. TODD, *Primal-dual interior-point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.
- [27] Y. E. NESTEROV AND M. TODD, *Self-scaled barriers and interior-point methods for convex programming*, Math. Oper. Res., 22 (1997), pp. 1–42.
- [28] F. A. POTRA AND R. SHENG, *A superlinearly convergent primal-dual infeasible-interior-point algorithm for semidefinite programming*, SIAM J. Optim., to appear.
- [29] J. F. STURM AND S. ZHANG, *Symmetric Primal-dual Path-following Algorithms for Semidefinite Programming*, Report 9554/A, Econometric Institute, Erasmus University, Rotterdam, The Netherlands, November 1995.
- [30] M. J. TODD, K. C. TOH, AND R. H. TÜTÜNCÜ, *On the Nesterov–Todd direction in semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 769–796.
- [31] P. TSENG, *Search directions and convergence analysis of some infeasible path-following methods for the monotone semi-definite LCP*, Optim. Meth. and Software, to appear.
- [32] L. VANDENBERGHE AND S. BOYD, *A primal-dual potential reduction method for problems involving matrix inequalities*, Math. Programming, 69 (1995), pp. 205–236.
- [33] Y. YE, *A class of projective transformations for linear programming*, SIAM J. Comput., 19 (1990), pp. 457–466.
- [34] Y. ZHANG, *On extending some primal-dual interior-point algorithms from linear programming to semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 365–386.

GLOBAL ANALYSIS OF THE DENNIS–WOLKOWICZ LEAST-CHANGE SECANT ALGORITHM*

LIXING HAN[†] AND GUANGHUI LIU[‡]

Abstract. This paper explores the global convergence of the least-change secant method proposed by Dennis and Wolkowicz [*SIAM J. Numer. Anal.*, 30 (1993), pp. 1291–1314]. This method is a member of the Broyden family, but it doesn't necessarily belong to the Broyden convex family. Furthermore, it can be very close to the DFP method. We will prove that this method, with the Wolfe line search on uniformly convex objective functions, is globally convergent.

Key words. Broyden family, DW algorithm, Wolfe line search, global convergence

AMS subject classifications. 90C30, 65K05

PII. S1052623495280378

1. Introduction. We consider quasi-Newton methods for the unconstrained optimization problem

$$\min_{x \in R^n} f(x),$$

where $f : R^n \rightarrow R^1$, $f \in C^2$. The quasi-Newton methods are often used in practice. The iterate x_{k+1} has the form

$$x_{k+1} = x_k + \lambda_k d_k,$$

where λ_k is a stepsize; d_k is a search direction and is determined by

$$d_k = -H_k g_k = -B_k^{-1} g_k,$$

where g_k denotes the gradient of f at the point x_k , and B_k is an approximation of the true Hessian $\nabla^2 f(x)$ at x_k . The matrix B_k (or its inverse matrix H_k) is generated by one of the iterative formulae of quasi-Newton methods such as the Broyden family

$$(1) \quad B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k} + (1 - \phi_k)(s_k^T B_k s_k) v_k v_k^T,$$

where

$$s_k = x_{k+1} - x_k = \lambda_k d_k, \quad y_k = g_{k+1} - g_k \\ v_k = \frac{y_k}{y_k^T s_k} - \frac{B_k s_k}{s_k^T B_k s_k},$$

and ϕ_k is a parameter. If $\phi_k = 0$, the above formula is called the DFP update. If $\phi_k = 1$ it is called the BFGS update. These two updates are the best known members

*Received by the editors January 25, 1995; accepted for publication (in revised form) June 6, 1997; published electronically July 20, 1998.

<http://www.siam.org/journals/siopt/8-3/28037.html>

[†]Department of Mathematics, U-9, University of Connecticut, Storrs, CT 06269 (lih95001@uconnvm.uconn.edu).

[‡]Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL 60208 (guanghui@pluto.ece.nwu.edu).

of this family (see [6, 9]). If $\phi_k \in [0, 1]$, it is called the Broyden convex family. If $\phi_k \in [\delta, 1]$, $\delta \in (0, 1]$, then the corresponding Broyden convex family is called the restricted Broyden convex family.

Many authors have investigated the convergence properties of the Broyden family (see, e.g., Byrd, Liu, and Nocedal [1]; Byrd and Nocedal [2]; Werner [16]; Zhang and Tewarson [18]; etc). In 1976, Powell [15] proved convergence properties of the BFGS method with inexact line searches. In 1987, Byrd, Nocedal, and Yuan [3] proved the convergence of the restricted Broyden convex family with the Wolfe line search on convex functions. They showed that the restricted Broyden convex family has the important property of self-correction. However, the DFP method doesn't possess such a property. It is still an open question whether the DFP method, with inexact line searches, on uniformly convex functions, converges globally (see Nocedal [10]). In 1988, Zhang and Tewarson [18] extended the global convergence result of [3] to a larger subfamily of the Broyden family of algorithms which allow $\phi_k > 1$. They show that if

$$1 - \phi_k \in [(1 - v)\phi_k^c, 1 - \delta],$$

where $v \in (0, 1)$, $\delta > 0$, and

$$\phi_k^c = \frac{1}{1 - u_k},$$

$$u_k = \frac{y_k^T H_k y_k s_k^T B_k s_k}{(y_k^T s_k)^2},$$

then the Broyden family of algorithms, with the Wolfe line search on convex functions, converges globally.

Methods in the Broyden family are secant-type methods; i.e., they have the property that the Hessian approximation B_{k+1} satisfies the secant condition

$$B_{k+1} s_k = y_k.$$

In [7], Dennis and Wolkowicz considered 14 secant-type methods, some known (see [4, 5, 8, 11, 12, 14, 17]) and some derived in their paper, and did extensive numerical experiments. Their experiments showed that the best among these 14 methods was method 10, which uses initial inverse sizing only and is a member of the Broyden family, with ϕ_k in (1) set to

$$(2) \quad \phi_k = \frac{1}{\left(\frac{b_k}{h_k} + 1 - \frac{b_k^2}{a_k h_k}\right)},$$

where

$$a_k = y_k^T H_k y_k, \quad b_k = y_k^T s_k, \quad h_k = s_k^T B_k s_k.$$

We, for the sake of simplicity, refer to method 10 as the Dennis–Wolkowicz (DW) algorithm. Since it is possible that $\phi_k > 1$ in (2), the DW algorithm doesn't necessarily belong to the Broyden convex family. Furthermore, if $\frac{h_k}{b_k} \ll 1$, then

$$\phi_k \approx 0.$$

Thus, the DW update can be very close to the DFP update. The purpose of this paper is to analyze the global convergence properties of this method. We will show that it still possesses the property of self-correction, on which we will base our proof of the global convergence of the DW algorithm with the commonly used Wolfe line search on uniformly convex functions.

The rest of the paper is organized as follows. In section 2, we present some preliminaries and assumptions. In section 3, we give some basic lemmas. In section 4, we derive some trace and determinant relations which show that the DW update possesses the self-correction property. Then, in section 5 we prove the global convergence of the DW method on uniformly convex functions. Finally, we present some discussion in section 6.

2. Preliminaries. In this paper, we analyze the DW algorithm with the following Wolfe line search:

$$\begin{aligned} f(x_k + \lambda_k d_k) &\leq f(x_k) + c_1 \lambda_k g_k^T d_k, \\ g(x_k + \lambda_k d_k)^T d_k &\geq c_2 g_k^T d_k, \end{aligned}$$

where $c_1 \in (0, 1)$ and $c_2 \in [c_1, 1)$. From the second inequality, we obtain

$$(3) \quad y_k^T s_k \geq (1 - c_2) \lambda_k (-g_k^T d_k) = (1 - c_2) (-g_k^T s_k),$$

which is heavily used in our later analysis.

We can compute the determinant of the matrix B_{k+1} from (1) (see Byrd, Liu, and Nocedal [1] and Pearson [13]):

$$\det(B_{k+1}) = \det(B_k) \left[\phi_k \frac{y_k^T s_k}{s_k^T B_k s_k} + (1 - \phi_k) \frac{y_k^T B_k^{-1} y_k}{y_k^T s_k} \right].$$

By using the definitions of ϕ_k , a_k , b_k , and h_k , we deduce

$$\begin{aligned} \det(B_{k+1}) &= \det(B_k) \left[\phi_k \frac{b_k}{h_k} + (1 - \phi_k) \frac{a_k}{b_k} \right] \\ &= \det(B_k) \frac{\phi_k a_k}{h_k} \left[\frac{b_k}{a_k} + \left(\frac{1}{\phi_k} - 1 \right) \frac{h_k}{b_k} \right] \\ &= \det(B_k) \frac{\phi_k a_k}{h_k} \left[\frac{b_k}{a_k} + \left(\frac{b_k}{h_k} - \frac{b_k^2}{a_k h_k} \right) \frac{h_k}{b_k} \right] \\ (4) \quad &= \det(B_k) \frac{\phi_k a_k}{h_k}. \end{aligned}$$

By using the definitions of u_k and ϕ_k^c , we can also deduce

$$\begin{aligned} \det(B_{k+1}) &= \det(B_k) \frac{b_k}{h_k} \left[\phi_k + (1 - \phi_k) \frac{a_k h_k}{b_k^2} \right] \\ &= \det(B_k) \frac{b_k}{h_k} [(\phi_k - 1) + (1 - \phi_k) u_k + 1] \\ &= \det(B_k) \frac{b_k}{h_k} [1 + (1 - \phi_k)(u_k - 1)] \\ &= \det(B_k) \frac{b_k}{h_k} \left[1 - \frac{(1 - \phi_k)}{\phi_k^c} \right]. \end{aligned}$$

It is known that if the initial Hessian approximation B_1 is selected to be symmetric positive definite, and if at each step $y_k^T s_k > 0$ and $1 - \phi_k > \phi_k^c$, then, by induction, all the matrices B_{k+1} remain symmetric positive definite (see Fletcher [9] and Byrd, Liu, and Nocedal [1]). Starting from any point x_1 and any symmetric positive definite matrix B_1 , we assume that B_k is symmetric positive definite, then a_k and h_k are positive, and since we use the Wolfe line search, from (3), we obtain that $y_k^T s_k > 0$. By the Cauchy–Schwarz inequality, we have

$$b_k^2 \leq a_k h_k.$$

Substituting this into (2), we obtain

$$(5) \quad \frac{h_k}{b_k} \geq \phi_k \geq \frac{1}{1 + \frac{b_k}{h_k}} > 0.$$

Therefore,

$$\frac{\phi_k a_k}{b_k} > 0.$$

From the above two derivations for $\det(B_{k+1})$ we see

$$\det(B_k) \frac{\phi_k a_k}{h_k} = \det(B_k) \frac{b_k}{h_k} \left[1 - \frac{(1 - \phi_k)}{\phi_k^c} \right],$$

which implies

$$1 - \frac{(1 - \phi_k)}{\phi_k^c} = \frac{\phi_k a_k}{b_k} > 0.$$

Since $u_k \geq 1$, we have $\phi_k^c \leq 0$. Therefore,

$$1 - \phi_k > \phi_k^c.$$

We know that all the matrices H_{k+1}, B_{k+1} ($k \geq 0$) are symmetric positive definite. We denote the trace of B_{k+1} by $\text{Tr}(B_{k+1})$; then $\forall k \geq 0$,

$$(6) \quad \text{Tr}(B_{k+1}) > 0.$$

From the iterative formula (1) of the matrix B_k we have (see Byrd, Nocedal, and Yuan [3] or Byrd, Liu, and Nocedal [1])

$$(7) \quad \text{Tr}(B_{k+1}) = \text{Tr}(B_k) - \phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} + \frac{\|y_k\|^2}{y_k^T s_k} + (1 - \phi_k) \frac{\|y_k\|}{y_k^T s_k} \frac{s_k^T B_k s_k}{y_k^T s_k} - (1 - \phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k}.$$

Equations (4) and (7) are very important in the convergence analysis of the DW method.

We use the following assumptions:

(H) (1) $f \in C^2$, f is bounded below.

(2) The level set $\Omega_0 = \{x \in R^n | f(x) \leq f(x_0)\}$ is a convex set, and there exist constants $c_3, c_4 > 0$ such that $c_3 \|z\|^2 \leq z^T \nabla^2 f(x) z \leq c_4 \|z\|^2 \forall x \in \Omega_0, \forall z \in R^n$.

By the mean value formula, we have

$$g(x) - g(y) = \int_0^1 G(y + t(x - y))(x - y)dt,$$

which gives

$$\begin{aligned} [g(x) - g(y)]^T(x - y) &= \int_0^1 (x - y)^T G(y + t(x - y))(x - y)dt, \\ \|g(x) - g(y)\|^2 &= \int_0^1 (x - y)^T G^2(y + t(x - y))(x - y)dt. \end{aligned}$$

From this and assumption (H), it is easy to derive the following properties:

(P1) Ω_0 is bounded.

(P2) $[g(x) - g(y)]^T(x - y) \geq c_3\|x - y\|^2 \quad \forall x, y \in \Omega_0$.

(P3) $c_4\|x - y\|^2 \geq [g(x) - g(y)]^T(x - y) \quad \forall x, y \in \Omega_0$.

(P4) $\|g(x) - g(y)\|^2 \leq c_4^2\|x - y\|^2 \leq \frac{c_4^2}{c_3}[g(x) - g(y)]^T(x - y) \quad \forall x, y \in \Omega_0$.

(P5) f has a unique minimizer x_* on Ω_0 , $g(x_*) = 0$.

Throughout this paper, we always assume that assumption (H) holds and that the steplength λ_k satisfies the Wolfe condition. The symbols $c_1, c_2, \dots, c_{24}$ always denote some positive constants.

3. Some basic lemmas. From (P1) and the Wolfe line search, we can assume that there exist positive constants c_5, c_6 such that

$$(8) \quad \|s_k\| \leq c_5, \quad \|g_k\| \leq c_6.$$

LEMMA 1. *Suppose that $\{x_k\}$ is generated by the DW algorithm, where the steplength λ_k satisfies the Wolfe line search. Assume that the level set Ω_0 is bounded. Then,*

$$\sum_{k=1}^{+\infty} (-g_k^T s_k) < +\infty.$$

Proof. From the first inequality of the Wolfe line search, we see

$$f(x_k) \geq f(x_{k+1}), \quad k \geq 1,$$

and

$$-g_k^T s_k \leq \frac{1}{c_1}[f(x_k) - f(x_{k+1})].$$

Since $x_k \in \Omega_0$, which is bounded, we have

$$\sum_{k=1}^{+\infty} (-g_k^T s_k) \leq \frac{1}{c_1} \sum_{k=1}^{+\infty} [f(x_k) - f(x_{k+1})] < +\infty. \quad \square$$

LEMMA 2. *Assume that assumption (H) holds and the steplength λ_k satisfies the Wolfe line search. Then, there exist two positive constants c_7, c_8 such that*

$$c_7(-g_k^T s_k) \leq \|s_k\|^2 \leq c_8(-g_k^T s_k).$$

Proof. On one hand, by (3) and (P3) we have

$$(1 - c_2)(-g_k^T s_k) \leq y_k^T s_k \leq c_4 \|s_k\|^2,$$

i.e.,

$$\frac{1 - c_2}{c_4}(-g_k^T s_k) \leq \|s_k\|^2.$$

On the other hand, by Taylor's theorem we have

$$f(x_{k+1}) = f(x_k) + g_k^T s_k + \frac{1}{2} s_k^T G(x_k + t_k s_k) s_k, \quad t_k \in (0, 1),$$

which, combined with assumption (H), gives

$$(9) \quad f(x_{k+1}) - f(x_k) - g_k^T s_k = \frac{1}{2} s_k^T G(x_k + t_k s_k) s_k \geq \frac{1}{2} c_3 \|s_k\|^2.$$

From (H) and the first inequality of the Wolfe line search, we have

$$f(x_{k+1}) - f(x_k) - g_k^T s_k \leq (c_1 - 1) g_k^T s_k;$$

this and (9) imply

$$\|s_k\|^2 \leq \frac{2(1 - c_1)}{c_3} (-g_k^T s_k).$$

Denote

$$c_7 = \frac{1 - c_2}{c_4}, \quad c_8 = \frac{2(1 - c_2)}{c_3}.$$

Thus, the lemma is true. $\square$

LEMMA 3. Assume assumption (H) holds. Suppose that $\{x_k\}$ is generated by the DW algorithm, where λ_k satisfies the Wolfe line search. Denote

$$\gamma_k = \frac{(-g_k^T d_k)}{\|d_k\|}.$$

Then,

$$\sum_{k=1}^{+\infty} (\gamma_k)^2 < +\infty.$$

Proof. Since $\gamma_k = \frac{-g_k^T s_k}{\|s_k\|}$, by Lemma 2 we have

$$(\gamma_k)^2 = \frac{(-g_k^T s_k)^2}{\|s_k\|^2} \leq \frac{(-g_k^T s_k)}{c_7}.$$

By Lemma 1, we see that this lemma is true. $\square$

LEMMA 4. Assume that all the assumptions of Lemma 3 hold. Then,

$$\lim_{k \rightarrow \infty} |y_k^T g_k| = 0.$$

Proof. From Lemmas 1 and 2, we obtain

$$\sum_{k=1}^{+\infty} \|s_k\|^2 < +\infty;$$

then,

$$\lim_{k \rightarrow \infty} \|s_k\| = 0.$$

By (P4) and (8), we have

$$|y_k^T g_k| \leq \|y_k\| \|g_k\| \leq c_4 \|s_k\| \|g_k\| \leq c_4 c_6 \|s_k\|.$$

Thus,

$$\lim_{k \rightarrow \infty} |y_k^T g_k| = 0. \quad \square$$

4. Some results on $\det(B_{k+1})$ and $\text{Tr}(B_{k+1})$. To prove the global convergence of the DW method, we need to derive some useful trace and determinant relations. We first derive the determinant relation.

THEOREM 5. *Assume that assumption (H) holds. Let x_1 be any starting point and let B_1 be any symmetric positive definite matrix. Suppose that $\{x_k\}$ is generated by the DW algorithm with the Wolfe line search. Then,*

$$\det(B_{k+1}) \geq \begin{cases} \frac{b_k}{2h_k} \det(B_k) & \text{if } \frac{b_k}{h_k} > 1, \\ \left(\frac{b_k}{2h_k}\right)^2 \det(B_k) & \text{if } \frac{b_k}{h_k} \leq 1. \end{cases}$$

Proof. By (4) we have

$$\det(B_{k+1}) = \det(B_k) \frac{b_k}{h_k} \frac{a_k}{b_k} \phi_k.$$

Since $b_k^2 \leq a_k h_k$,

$$\frac{a_k}{b_k} \geq \frac{b_k}{h_k}.$$

Thus,

$$(10) \quad \det(B_{k+1}) \geq \left(\frac{b_k}{h_k}\right)^2 \phi_k \det(B_k).$$

We rewrite ϕ_k as

$$(11) \quad \phi_k = \frac{h_k}{b_k} \frac{1}{1 + \frac{h_k}{b_k} \left(1 - \frac{b_k^2}{a_k h_k}\right)}.$$

Note that

$$0 \leq 1 - \frac{b_k^2}{a_k h_k} < 1.$$

If $\frac{b_k}{h_k} > 1$, then by (11) we have

$$\phi_k \geq \frac{h_k}{b_k} \frac{1}{1 + (1 - \frac{b_k^2}{a_k h_k})} \geq \frac{h_k}{2b_k}.$$

This and (10) imply

$$\det(B_{k+1}) \geq \left(\frac{b_k}{h_k}\right)^2 \frac{h_k}{2b_k} \det(B_k) = \frac{b_k}{2h_k} \det(B_k).$$

If $\frac{b_k}{h_k} \leq 1$, then by (11) we have

$$\phi_k \geq \frac{h_k}{b_k} \frac{1}{1 + \frac{h_k}{b_k}} \geq \frac{1}{2}.$$

This and (10) give

$$\det(B_{k+1}) \geq \frac{1}{2} \left(\frac{b_k}{h_k}\right)^2 \det(B_k) \geq \left(\frac{b_k}{2h_k}\right)^2 \det(B_k).$$

Therefore, this theorem is true. $\square$

In the following analyses, we focus on deriving a trace relation between $\text{Tr}(B_{k+1})$ and $\text{Tr}(B_k)$, which is crucial to the analysis of global convergence. We will compare the fourth and fifth terms with the second term of (7).

LEMMA 6. *If $\phi_k \geq 1$, then $|1 - \phi_k| \leq \phi_k$. If $0 < \phi_k < 1$, then $|1 - \phi_k| \leq \frac{b_k}{h_k} \phi_k$.*

Proof. If $\phi_k \geq 1$, then

$$|1 - \phi_k| = \phi_k - 1 < \phi_k.$$

If $0 < \phi_k < 1$, then

$$|1 - \phi_k| = 1 - \phi_k = \frac{\frac{b_k}{h_k} - \frac{b_k^2}{a_k h_k}}{\frac{b_k}{h_k} + 1 - \frac{b_k^2}{a_k h_k}} = \left(\frac{b_k}{h_k} - \frac{b_k^2}{a_k h_k}\right) \phi_k \leq \frac{b_k}{h_k} \phi_k. \quad \square$$

We now compare the fifth term with the second term of (7).

THEOREM 7. *Suppose that $\{x_k\}$ is generated by the DW algorithm with the Wolfe line search. Assume that there exists a constant $c_9 > 0$ such that*

$$(12) \quad \|g_k\| \geq c_9.$$

Then, there exists a positive constant c_{10} such that

$$\left| (1 - \phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| \leq \max \left\{ c_{10}, \frac{\phi_k}{4} \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} \right\}$$

holds for sufficiently large k .

Proof. By (3) and (12), we have

$$\begin{aligned}
 \left| \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| &\leq \frac{2\lambda_k |y_k^T g_k|}{(1-c_2)(-g_k^T s_k)} \\
 &= \frac{2\lambda_k^2 |y_k^T g_k|}{(1-c_2)s_k^T B_k s_k} \\
 &= \frac{2\|B_k s_k\|^2 |y_k^T g_k|}{(1-c_2)\|g_k\|^2 s_k^T B_k s_k} \\
 &\leq \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} \frac{2|y_k^T g_k|}{(1-c_2)c_9^2}.
 \end{aligned}$$

Hence, by Lemma 4, we know that for sufficiently large k ,

$$(13) \quad \left| \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| \leq \frac{1}{8} \frac{\|B_k s_k\|^2}{s_k^T B_k s_k}.$$

Now, our proof is divided into two cases.

Case 1. $\phi_k \geq 1$.

In this case, by Lemma 6 and (13), we have that, for sufficiently large k ,

$$(14) \quad \left| (1-\phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| \leq \frac{|\phi_k|}{4} \frac{\|B_k s_k\|^2}{s_k^T B_k s_k}.$$

Case 2. $0 < \phi_k < 1$.

In this case, $|1-\phi_k| < 1$; then

$$\begin{aligned}
 \left| (1-\phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| &\leq \left| \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| \\
 (15) \quad &= \phi_k \frac{1}{\phi_k} \left| \frac{2y_k^T B_k s_k}{y_k^T s_k} \right|.
 \end{aligned}$$

By (5),

$$\frac{1}{\phi_k} \leq \frac{b_k}{h_k} + 1.$$

Substituting this into (15), we obtain

$$\left| (1-\phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| \leq \phi_k \left(\frac{b_k}{h_k} + 1 \right) \left| \frac{2y_k^T B_k s_k}{y_k^T s_k} \right|.$$

Substituting (13) into this, we have that, for sufficiently large k ,

$$\begin{aligned}
 \left| (1-\phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| &\leq \phi_k \frac{b_k}{h_k} \left| \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| + \frac{\phi_k}{8} \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} \\
 (16) \quad &= \xi_k + \frac{\phi_k}{8} \frac{\|B_k s_k\|^2}{s_k^T B_k s_k},
 \end{aligned}$$

where

$$\xi_k = \phi_k \frac{b_k}{h_k} \left| \frac{2y_k^T B_k s_k}{y_k^T s_k} \right|.$$

We now divide Case 2 into two subcases.

Subcase 2.1. If $\xi_k \leq \frac{\phi_k \|B_k s_k\|^2}{8 s_k^T B_k s_k}$, then by (16) we immediately obtain

$$(17) \quad \left| (1 - \phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| \leq \frac{\phi_k \|B_k s_k\|^2}{4 s_k^T B_k s_k}.$$

Subcase 2.2. If $\xi_k > \frac{\phi_k \|B_k s_k\|^2}{8 s_k^T B_k s_k}$, then

$$\frac{b_k}{h_k} \left| \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| > \frac{1}{8} \frac{\|B_k s_k\|^2}{s_k^T B_k s_k},$$

i.e.,

$$16|y_k^T B_k s_k| > \|B_k s_k\|^2 = \lambda_k^2 \|g_k\|^2.$$

Substituting (12) into this, we obtain

$$|y_k^T B_k s_k| \geq c_{11} \lambda_k^2,$$

where $c_{11} = c_9^2/16$. From this we have

$$|y_k^T B_k s_k| \leq \frac{|y_k^T B_k s_k|^2}{c_{11} \lambda_k^2} = \frac{|y_k^T g_k|^2}{c_{11}};$$

from this, (P4), and (8), we have

$$|y_k^T B_k s_k| \leq \frac{\|y_k\|^2 \|g_k\|^2}{c_{11}} \leq \frac{c_4^2 \|s_k\|^2 c_6^2}{c_{11}} = c_{12} \|s_k\|^2,$$

where $c_{12} = c_4^2 c_6^2 / c_{11}$. From this and (P2), we have

$$(18) \quad \begin{aligned} \left| (1 - \phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| &\leq \left| \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| \\ &\leq \frac{|2y_k^T B_k s_k|}{c_3 \|s_k\|^2} \\ &\leq \frac{2c_{12}}{c_3} = c_{10}. \end{aligned}$$

Summarizing (14), (17), and (18), we complete the proof. $\square$

We now compare the fourth term with the second term of (7).

THEOREM 8. *Suppose that $\{x_k\}$ is generated by the DW algorithm with the Wolfe line search. Assume that (12) holds. Then, there exists a positive constant c_{13} such that*

$$\left| (1 - \phi_k) \frac{s_k^T B_k s_k}{y_k^T s_k} \right| \leq c_{13} \left[1 + \phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} (-g_k^T s_k) \right].$$

Proof. By (3) we deduce

$$\begin{aligned} \frac{s_k^T B_k s_k}{y_k^T s_k} &\leq \frac{s_k^T B_k s_k}{(1 - c_2)(-g_k^T s_k)} \\ &= \frac{\lambda_k}{(1 - c_2)s_k^T B_k s_k} (s_k^T B_k s_k) \\ &= \frac{\lambda_k^2}{(1 - c_2)s_k^T B_k s_k} (-g_k^T s_k) \\ &= \frac{\|B_k s_k\|^2}{(1 - c_2)\|g_k\|^2 s_k^T B_k s_k} (-g_k^T s_k). \end{aligned}$$

Substituting (12) into this, we have

$$(19) \quad \frac{s_k^T B_k s_k}{y_k^T s_k} \leq c_{14} \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} (-g_k^T s_k),$$

where $c_{14} = 1/[(1 - c_2)c_9^2]$.

Now, our analysis is divided into two cases.

Case 1. $0 < \phi_k < 1$.

In this case, by Lemma 6, we immediately obtain

$$\left| (1 - \phi_k) \frac{s_k^T B_k s_k}{y_k^T s_k} \right| \leq \phi_k \frac{b_k}{h_k} \left| \frac{s_k^T B_k s_k}{y_k^T s_k} \right| = \phi_k < 1.$$

Case 2. $\phi_k \geq 1$.

In this case, by Lemma 6 and (19), we have

$$\left| (1 - \phi_k) \frac{s_k^T B_k s_k}{y_k^T s_k} \right| \leq \phi_k \frac{s_k^T B_k s_k}{y_k^T s_k} \leq \phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} (-c_{14} g_k^T s_k).$$

Summarizing these two cases, we have

$$\left| \frac{(1 - \phi_k) s_k^T B_k s_k}{y_k^T s_k} \right| \leq c_{13} \left[1 + \phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} (-g_k^T s_k) \right].$$

Selecting $c_{13} = \max\{1, c_{14}\}$, we see that the theorem is true. $\square$

We can now derive the trace relation between $\text{Tr}(B_{k+1})$ and $\text{Tr}(B_k)$.

THEOREM 9. *Assume that assumption (H) and inequality (12) hold. Let x_1 be any starting point and B_1 be any symmetric positive definite matrix. Suppose that $\{x_k\}$ is generated by the DW algorithm with the Wolfe line search. Then there exists a constant $c_{15} > 0$ and an index $k_2 > 0$ such that $\forall k \geq k_2$,*

$$\text{Tr}(B_{k+1}) \leq \text{Tr}(B_k) - \frac{\phi_k}{2} \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} + c_{15}.$$

Proof. By (7)

$$(20) \quad \begin{aligned} \text{Tr}(B_{k+1}) &\leq \text{Tr}(B_k) - \phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} + \frac{\|y_k\|^2}{y_k^T s_k} \\ &\quad + \left| (1 - \phi_k) \frac{\|y_k\|^2}{y_k^T s_k} \frac{s_k^T B_k s_k}{y_k^T s_k} \right| + \left| (1 - \phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right|. \end{aligned}$$

From (P2) and (P4) we have

$$\frac{\|y_k\|^2}{y_k^T s_k} \leq \frac{c_4^2 \|s_k\|^2}{c_3 \|s_k\|^2} = c_{16},$$

where $c_{16} = c_4^2/c_3$. Substituting this into (20), we obtain

$$(21) \quad \begin{aligned} \text{Tr}(B_{k+1}) &\leq \text{Tr}(B_k) - \phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} + c_{16} \\ &\quad + c_{16} \left| (1 - \phi_k) \frac{s_k^T B_k s_k}{y_k^T s_k} \right| + \left| (1 - \phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right|. \end{aligned}$$

Using Theorem 8, we obtain

$$(22) \quad \begin{aligned} \text{Tr}(B_{k+1}) &\leq \text{Tr}(B_k) - \phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} + c_{16} \\ &\quad + c_{16} c_{13} \left[1 + \phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} (-g_k^T s_k) \right] + \left| (1 - \phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| \\ &= \text{Tr}(B_k) - [1 - c_{16} c_{13} (-g_k^T s_k)] \phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} + c_{16} \\ &\quad + c_{16} c_{13} + \left| (1 - \phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right|. \end{aligned}$$

By Theorem 7, we know that there exists $k_1 > 0$ such that $\forall k \geq k_1$,

$$\left| (1 - \phi_k) \frac{2y_k^T B_k s_k}{y_k^T s_k} \right| \leq c_{10} + \frac{\phi_k}{4} \frac{\|B_k s_k\|^2}{s_k^T B_k s_k}.$$

Substituting this into (22), we have

$$\text{Tr}(B_{k+1}) \leq \text{Tr}(B_k) - \left[\frac{3}{4} - c_{16} c_{13} (-g_k^T s_k) \right] \phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} + c_{16} + c_{16} c_{13} + c_{10}.$$

By Lemma 1, we have

$$\lim_{k \rightarrow \infty} \left[\frac{3}{4} - c_{16} c_{13} (-g_k^T s_k) \right] = \frac{3}{4} > \frac{1}{2}.$$

Thus, there exists $k_2 > 0$ such that $\forall k \geq k_2$,

$$\text{Tr}(B_{k+1}) \leq \text{Tr}(B_k) - \frac{1}{2} \phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} + c_{15},$$

where $c_{15} = c_{16} + c_{16} c_{13} + c_{10}$. $\square$

From Theorems 5–9, we can see the property of self-correction of the DW algorithm. Indeed, the trace relation in Theorem 9 shows that the eigenvalues of the matrices B_k cannot become too large, and the determinant relation in Theorem 5

shows that they cannot become too small. This is used in section 5 to show the global convergence of the DW method.

5. Convergence analysis. In this section, we will prove the global convergence of the DW algorithm:

$$\lim_{k \rightarrow \infty} \inf \|g_k\| = 0.$$

Our approach is to assume by contradiction that

$$\lim_{k \rightarrow \infty} \inf \|g_k\| > 0;$$

i.e., we assume (12) holds, from which we will derive a contradiction. Our analysis will be developed by discussing two cases: either $b_k > h_k$ or $b_k \leq h_k$. Thus, we define

$$p(j) = \begin{cases} 1 & \text{if } b_j > h_j, \\ 2 & \text{if } b_j \leq h_j. \end{cases}$$

LEMMA 10. *Assume that all the assumptions of Theorem 9 hold. Then, there exist positive constants c_{17}, c_{18} such that*

$$\sum_{j=1}^k \left(\frac{2h_j}{b_j} \right)^{p(j)/2} \geq c_{17}k,$$

$$\sum_{j=1}^k \left(\frac{2h_j}{b_j} \right)^{2p(j)} \geq c_{18}k.$$

Proof. By Theorem 9, we have that there exists a constant $c_{19} > 0$ such that

$$\text{Tr}(B_{k+1}) \leq c_{19}k;$$

this and the geometric-arithmetic mean inequality give

$$(23) \quad \det(B_{k+1}) \leq \left(\frac{\text{Tr}(B_{k+1})}{n} \right)^n \leq \left(\frac{c_{19}k}{n} \right)^n.$$

By Theorem 5 we have

$$(24) \quad \begin{aligned} \det(B_{k+1}) &\geq \left(\frac{b_k}{2h_k} \right)^{p(k)} \det(B_k) \\ &\geq \dots \\ &\geq \left[\prod_{j=1}^k \left(\frac{b_j}{2h_j} \right)^{p(j)} \right] \det(B_1). \end{aligned}$$

Considering (23), we have

$$\prod_{j=1}^k \left(\frac{2h_j}{b_j} \right)^{p(j)} \geq \frac{n^n \det(B_1)}{(c_{19}k)^n} \geq (c_{20})^k,$$

where $c_{20} > 0$, c_{20} is a small constant. From this we obtain

$$\prod_{j=1}^k \left(\frac{2h_j}{b_j} \right)^{\frac{p(j)}{2}} \geq (c_{17})^k,$$

$$\prod_{j=1}^k \left(\frac{2h_j}{b_j} \right)^{2p(j)} \geq (c_{18})^k,$$

where $c_{17} = \sqrt{c_{20}}$, $c_{18} = c_{20}^2$. From these two inequalities, we use the geometric-arithmetic mean inequality again; then the proof of this lemma is complete. $\square$

LEMMA 11. Assume that all the assumptions of Theorem 9 hold. Then, there exists a positive constant c_{21} such that

$$\phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} \geq \frac{c_{21}}{\gamma_k} \left(\frac{h_k}{b_k} \right)^{3-p(k)},$$

where $\gamma_k = -g_k^T d_k / \|d_k\|$.

Proof. By (12), (8), and (3), we deduce

$$\begin{aligned} \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} &= \frac{\|g_k\|^2}{-g_k^T d_k} \\ &\geq \frac{c_9^2}{-g_k^T d_k} \\ &= \frac{c_9^2 \lambda_k}{-g_k^T s_k} \\ &= \frac{c_9^2 s_k^T B_k s_k}{(-g_k^T s_k)^2} \\ &= \frac{c_9^2 s_k^T B_k s_k}{(-g_k^T s_k) \gamma_k \|s_k\|} \\ &\geq \frac{c_9^2 s_k^T B_k s_k}{(-g_k^T s_k) \gamma_k c_5} \\ &\geq \frac{c_9^2 (1 - c_2) s_k^T B_k s_k}{(g_k^T s_k) \gamma_k c_5} \\ (25) \quad &= c_{22} \frac{h_k}{b_k \gamma_k}, \end{aligned}$$

where $c_{22} = c_9^2 (1 - c_2) / c_5$.

Now, we divide our proof into two cases.

Case 1. $h_k < b_k$.

In this case, $p(k) = 1$, and, from (5), we have

$$\phi_k \geq \frac{1}{1 + b_k/h_k} \geq \frac{1}{b_k/h_k + b_k/h_k} = \frac{h_k}{2b_k}.$$

By (25), we have

$$\phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} \geq \frac{c_{22}}{2\gamma_k} \left(\frac{h_k}{b_k} \right)^2 = \frac{c_{22}}{2\gamma_k} \left(\frac{h_k}{b_k} \right)^{3-p(k)}.$$

Case 2. $h_k \geq b_k$.

In this case, $p(k) = 2$, and

$$\phi_k \geq \frac{1}{1 + b_k/h_k} \geq \frac{1}{2}.$$

Combining this with (25), we have

$$\phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} \geq \frac{c_{22}}{2\gamma_k} \frac{h_k}{b_k} = \frac{c_{22}}{2\gamma_k} \left(\frac{h_k}{b_k}\right)^{3-p(k)}.$$

Summarizing these two cases, we have

$$\phi_k \frac{\|B_k s_k\|^2}{s_k^T B_k s_k} \geq \frac{c_{21}}{\gamma_k} \left(\frac{h_k}{b_k}\right)^{3-p(k)},$$

where $c_{21} = c_{22}/2$. Therefore, this lemma is true. $\square$

Now, we prove the global convergence of the DW algorithm.

THEOREM 12. *Assume that assumption (H) holds. Let x_1 be any starting point and B_1 be any symmetric positive definite matrix. Suppose that $\{x_k\}$ is generated by the DW algorithm with the Wolfe line search. Then*

$$\liminf_k \|g_k\| = 0.$$

Proof. Suppose that

$$\liminf_k \|g_k\| > 0;$$

i.e., (12) holds. Therefore, Theorems 5–9 in the preceding section hold.

Denote

$$\begin{aligned} N_k &= \{1, 2, \dots, k\}, \\ I_k &= \{j | h_j \geq b_j, j \in N_k\}, \\ J_k &= N_k - I_k, \end{aligned}$$

with $|I_k|$ being the number of elements in I_k and $|J_k|$ being the number of elements in J_k .

By Lemma 10, we have

$$\sum_{j=1}^k \left(\frac{2h_j}{b_j}\right)^{\frac{p(j)}{2}} \geq c_{17}k.$$

Since $\frac{p(j)}{2} \leq 1$, we obtain

$$\sum_{j=1}^k \left(\frac{h_j}{b_j}\right)^{\frac{p(j)}{2}} \geq \frac{c_{17}}{2}k.$$

Considering the definitions of I_k , J_k , and $p(j)$, we partition the left-hand side terms of the inequality into two parts:

$$\sum_{j \in I_k} \frac{h_j}{b_j} + \sum_{j \in J_k} \left(\frac{h_j}{b_j}\right)^{\frac{1}{2}} \geq \frac{c_{17}}{2}k,$$

from which we know that either

$$(26) \quad \sum_{j \in I_k} \frac{h_j}{b_j} \geq \frac{c_{17}}{4} k$$

or

$$(27) \quad \sum_{j \in J_k} \left(\frac{h_j}{b_j} \right)^{\frac{1}{2}} \geq \frac{c_{17}}{4} k.$$

Now, we divide our proof into two cases.

Case 1. Inequality (26) holds.

In this case, by Lemma 3, we know that there exists $k_3 > 0$ such that $\forall j \geq k_3$,

$$\frac{1}{\gamma_j} \geq \frac{16c_{15}}{c_{21}c_{17}};$$

from this and Lemma 11 we obtain that $\forall j \geq k_3$,

$$\phi_j \frac{\|B_j s_j\|^2}{s_j^T B_j s_j} \geq \frac{c_{21}}{\gamma_k} \left(\frac{h_j}{b_j} \right)^{3-p(k)} \geq \frac{16c_{15}}{c_{17}} \left(\frac{h_j}{b_j} \right)^{3-p(j)}.$$

Thus, $\forall j \in I_k, j \geq k_3$,

$$\phi_j \frac{\|B_j s_j\|^2}{s_j^T B_j s_j} \geq \frac{16c_{15}}{c_{17}} \frac{h_j}{b_j}.$$

By Theorem 9, we deduce from (26) that $\forall k \geq k_3$,

$$\begin{aligned} \text{Tr}(B_{k+1}) &\leq \text{Tr}(B_{k_3}) + c_{15}(k - k_3 + 1) - \sum_{j=k_3}^k \frac{\phi_j}{2} \frac{\|B_j s_j\|^2}{s_j^T B_j s_j} \\ &\leq \text{Tr}(B_{k_3}) + c_{15}(k - k_3 + 1) - \sum_{j \in I_k, j \geq k_3} \frac{\phi_j}{2} \frac{\|B_j s_j\|^2}{s_j^T B_j s_j} \\ &\leq \text{Tr}(B_{k_3}) + c_{15}(k - k_3 + 1) - \sum_{j \in I_k, j \geq k_3} \frac{8c_{15}}{c_{17}} \frac{h_j}{b_j} \\ &= \text{Tr}(B_{k_3}) + c_{15}(k - k_3 + 1) - \frac{8c_{15}}{c_{17}} \sum_{j \in I_k} \frac{h_j}{b_j} + \sum_{j \in I_k, j < k_3} \frac{8c_{15}}{c_{17}} \frac{h_j}{b_j} \\ &\leq \text{Tr}(B_{k_3}) + c_{15}(k - k_3 + 1) - 2c_{15}k + \sum_{j \in I_k, j < k_3} \frac{8c_{15}}{c_{17}} \frac{h_j}{b_j}. \end{aligned}$$

Substituting (6) into this, we have that $\forall k > k_3$,

$$c_{15}k \leq \text{Tr}(B_{k_3}) + c_{15}(1 - k_3) + \sum_{j \in I_k, j < k_3} \frac{8c_{15}}{c_{17}} \frac{h_j}{b_j}.$$

This is a contradiction, since the right-hand side of the inequality is bounded.

Case 2. Inequality (27) holds.

In this case, we denote

$$J_{k,1} = \left\{ j | j \in J_k, \left(\frac{h_j}{b_j} \right)^{\frac{1}{2}} < \frac{c_{17}}{8} \right\},$$

$$J_{k,2} = \left\{ j | j \in J_k, \left(\frac{h_j}{b_j} \right)^{\frac{1}{2}} \geq \frac{c_{17}}{8} \right\},$$

with $|J_{k,1}|$ being the number of elements of $J_{k,1}$ and $|J_{k,2}|$ being the number of elements of $J_{k,2}$.

Since $h_j < b_j$, for all $j \in J_k$, and

$$J_k = J_{k,1} \cup J_{k,2}, \quad |J_{k,1}| \leq |J_k| \leq k,$$

from (27) we have

$$\begin{aligned} \frac{c_{17}}{4}k &\leq \sum_{j \in J_{k,1}} \left(\frac{h_j}{b_j} \right)^{\frac{1}{2}} + \sum_{j \in J_{k,2}} \left(\frac{h_j}{b_j} \right)^{\frac{1}{2}} \\ &\leq \sum_{j \in J_{k,1}} \frac{c_{17}}{8} + \sum_{j \in J_{k,2}} 1 \\ &\leq \frac{c_{17}}{8}|J_{k,1}| + |J_{k,2}| \\ (28) \quad &\leq \frac{c_{17}}{8}k + |J_{k,2}|. \end{aligned}$$

Thus,

$$|J_{k,2}| \geq \frac{c_{17}}{8}k.$$

Since $\forall j \in J_{k,2}$,

$$\frac{h_j}{b_j} \geq \frac{c_{17}^2}{64},$$

from this and Lemma 11 we obtain that $\forall j \in J_{k,2}$,

$$(29) \quad \phi_j \frac{\|B_j s_j\|^2}{s_j^T B_j s_j} \geq \frac{c_{21}}{\gamma_j} \left(\frac{h_j}{b_j} \right)^2 \geq c_{23} \frac{1}{\gamma_j},$$

where

$$c_{23} = \frac{c_{21}c_{17}^4}{4096}.$$

By (29) and Lemma 3, there exists $k_4 > k_2$ such that $\forall j \geq k_4$,

$$(30) \quad \phi_j \frac{\|B_j s_j\|^2}{s_j^T B_j s_j} \geq \frac{32c_{15}}{c_{17}}.$$

From Theorem 9, (30), and (28), we deduce that $\forall k > k_4$,

$$\begin{aligned}
\text{Tr}(B_{k+1}) &\leq \text{Tr}(B_{k_4}) + c_{15}(k - k_4 + 1) - \sum_{j \in J_{k,2}, j \geq k_4} \frac{\phi_j \|B_j s_j\|^2}{2 s_j^T B_j s_j} \\
&\leq \text{Tr}(B_{k_4}) + c_{15}(k - k_4 + 1) - \sum_{j \in J_{k,2}, j \geq k_4} \frac{16c_{15}}{c_{17}} \\
&= \text{Tr}(B_{k_4}) + c_{15}(k - k_4 + 1) - \sum_{j \in J_{k,2}} \frac{16c_{15}}{c_{17}} + \sum_{j \in J_{k,2}, j \leq k_4-1} \frac{16c_{15}}{c_{17}} \\
&= \text{Tr}(B_{k_4}) + c_{15}(k - k_4 + 1) - |J_{k,2}| \frac{16c_{15}}{c_{17}} + \sum_{j \in J_{k,2}, j \leq k_4-1} \frac{16c_{15}}{c_{17}} \\
&\leq \text{Tr}(B_{k_4}) + c_{15}(k - k_4 + 1) - \frac{c_{17}}{8} k \left(\frac{16c_{15}}{c_{17}} \right) + \sum_{j \in J_{k,2}, j \leq k_4-1} \frac{16c_{15}}{c_{17}} \\
&= \text{Tr}(B_{k_4}) - c_{15}k + c_{15}(1 - k_4) + \sum_{j \in J_{k,2}, j \leq k_4-1} \frac{16c_{15}}{c_{17}}.
\end{aligned}$$

Substituting (6) into the above inequality, we have that $\forall k > k_4$,

$$k \leq \frac{\text{Tr}(B_{k_4})}{c_{15}} + (1 - k_4) + \sum_{j \in J_{k,2}, j \leq k_4-1} \left(\frac{16c_{15}}{c_{17}} \right).$$

This is a contradiction.

From Cases 1 and 2, we conclude

$$\liminf_k \|g_k\| = 0. \quad \square$$

Under assumption (H), we can easily derive the following corollary from Theorem 12.

COROLLARY 13. *Assume that all the assumptions of Theorem 5 hold. Then*

$$\lim_k x_k = x_*,$$

where x_* is the unique minimizer of $f(x)$ in Ω_0 .

Proof. Since f is convex on Ω_0 ,

$$f(x_*) - f(x_k) \geq g_k^T(x_* - x_k).$$

Thus, by assumption (H), there exists $c_{24} > 0$, such that

$$(31) \quad 0 \leq f(x_k) - f(x_*) \leq g_k^T(x_k - x_*) \leq \|g_k\| \|x_k - x_*\| \leq c_{24} \|g_k\|.$$

By Theorem 12, there exists a set $K \subseteq \{1, 2, \dots\}$ such that

$$\lim_{k \in K} \|g_k\| = 0;$$

from this and (31), we obtain

$$\lim_{k \in K} f(x_k) = f(x_*).$$

Note that the whole sequence $\{f(x_k)\}$ is convergent. Hence

$$(32) \quad \lim_{k \rightarrow \infty} f(x_k) = f(x_*).$$

From assumption (H), we also have

$$f(x_{k+1}) - f(x_*) \geq \frac{1}{2} c_3 \|x_{k+1} - x_*\|^2;$$

this and (32) imply

$$\lim_{k \rightarrow \infty} x_k = x_*. \quad \square$$

6. Discussion. We have studied the global convergence of the DW method. Specifically, for uniformly convex functions, we have shown that it is globally convergent with the Wolfe line search. To establish this global convergence result, we have relied heavily on the trace and determinant expressions for the matrices B_k . By making full use of these expressions, we have obtained useful properties of the DW algorithm, which hold even when the DW update goes beyond the Broyden convex family and approximates the DFP update.

The numerical results of Dennis and Wolkowicz [7] show that the DW method performs better than the BFGS method without sizing. It also outperforms the BFGS method with initial sizing which is believed to be among the best variable metric methods.

As argued by Byrd, Liu, and Nocedal [1] and Byrd, Nocedal, and Yuan [3], a good member of the Broyden family should possess the property that it can correct both large and small eigenvalues of the approximation matrix B_k efficiently. They call this property self-correction. In order to correct large eigenvalues efficiently, the term $1 - \phi_k$ should be as small as possible. However, to deal with small eigenvalues, one should ensure that $1 - \phi_k$ is not too close to ϕ_k^c . Thus, it seems that $1 - \phi_k = 0$ is a good choice, i.e., the BFGS update.

Now, we consider the DW method. Its parameter ϕ_k is given in (2). We compare the values of a_k , b_k , and h_k .

(i) If $h_k/b_k \ll 1$, this means that some eigenvalues of B_k are very small. Note that in this case $\phi_k \approx 0$. The DW method is close to the DFP method, where the eigenvalues of B_{k+1} of the DW method tend to increase more than those of the BFGS method.

(ii) If $h_k/b_k \gg 1$, this means that some of the eigenvalues of B_k are large. From (2) we see that, in this case, $\phi_k > 1$. Thus, the eigenvalues of B_{k+1} of the DW method tend to decrease more than those of the BFGS method. This is particularly important for the success of the DW method since the BFGS method suffers more from large eigenvalues than from small ones (see Byrd, Liu, and Nocedal [1]).

(iii) If $h_k/b_k = O(1)$, this generally means that the eigenvalues of B_k are moderate. In this case, we note that $\phi_k = O(1)$. The DW method is not much different from the BFGS method. Thus, it can correct the eigenvalues of B_k almost as efficiently as the BFGS method.

Summarizing the above three cases, we conclude that the DW method can usually correct very large and small eigenvalues of B_k more efficiently than the BFGS method. So, it is reasonable that the DW method works better than the BFGS method, as Dennis and Wolkowicz observed.

Acknowledgments. We are very grateful to the associate editor, Professor M. El-Alem, and the two anonymous referees for their constructive and insightful comments and suggestions. Also, we appreciate very much their patience in helping us revise the early versions of this paper. The second author thanks Professor Jorge Nocedal of Northwestern University for his great support and supervision.

REFERENCES

- [1] R. H. BYRD, D. C. LIU, AND J. NOCEDAL, *On the behavior of Broyden's class of quasi-Newton methods*, SIAM J. Optim., 2 (1992), pp. 533–557.
- [2] R. H. BYRD AND J. NOCEDAL, *A tool for the analysis of quasi-Newton methods with application to unconstrained minimization*, SIAM J. Numer. Anal., 26 (1989), pp. 727–739.
- [3] R. H. BYRD, J. NOCEDAL, AND Y. YUAN, *Global convergence of a class of quasi-Newton methods on convex problems*, SIAM J. Numer. Anal., 24 (1987), pp. 1171–1191.
- [4] W. C. DAVIDON, *Optimally conditioned optimization algorithms without line searches*, Math. Programming, 9 (1975), pp. 1–30.
- [5] J. E. DENNIS, JR., D. M. GAY, AND R. E. WELSCH, *An adaptive nonlinear least-squares algorithm*, ACM Trans. Math. Software, 7 (1981), pp. 348–368.
- [6] J. E. DENNIS AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1983. Mir Publishing Office, Moscow, 1988 (in Russian).
- [7] J. E. DENNIS, JR. AND H. WOLKOWICZ, *Sizing and least-change secant methods*, SIAM J. Numer. Anal., 30 (1993), pp. 1291–1314.
- [8] R. FLETCHER, *A new variational result for quasi-Newton formulae*, SIAM J. Optim., 1 (1991), pp. 18–21.
- [9] R. FLETCHER, *Practical Methods of Optimization*, 2nd ed., John Wiley, New York, 1987.
- [10] J. NOCEDAL, *Theory of algorithms for unconstrained optimization*, in Acta Numerica, A. Iserles, ed., Cambridge University Press, Cambridge, England, 1992, pp. 199–242.
- [11] S. S. OREN AND D. G. LUENBERGER, *Self-scaling variable metric (SSVM) algorithms, part I. Criteria and sufficient conditions for scaling a class of algorithms*, Management Sci., 20 (1974), pp. 845–862.
- [12] S. S. OREN AND E. SPEDICATO, *Optimal conditioning of self-scaling variable metric algorithms*, Math. Programming, 10 (1976), pp. 70–90.
- [13] J. D. PEARSON, *Variable metric methods of minimization*, The Computer J., 12 (1969), pp. 171–178.
- [14] K. H. PHUA AND D. F. SHANNO, *Numerical comparison of several variable metric algorithms*, J. Optim. Theory Appl., 25 (1978), pp. 507–518.
- [15] M. J. D. POWELL, *Some global convergence properties of a variable metric algorithm for minimization without exact line searches*, in Nonlinear Programming, SIAM-AMS Proceedings, Vol. IX, R. W. Cottle and C. E. Lemke, eds., AMS, Providence, RI, 1976.
- [16] J. WERNER, *Global Convergence of Quasi-Newton Methods with Practical Line Searches*, Technical Report, NAM-Bericht 67, Institut für Numerische und Angewandte Mathematik der Universität Göttingen, Germany, 1989.
- [17] H. WOLKOWICZ, *Measures for Symmetric Rank-one Updates*, Tech. Report CORR 90-03, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, ON, Canada, 1990.
- [18] Y. ZHANG AND R. P. TEWARSON, *Quasi-Newton algorithms with updates from the pre-convex part of Broyden's family*, IMA J. Numer. Anal., 8 (1988), pp. 487–509.

ON THE COMPLEXITY OF A PRACTICAL INTERIOR-POINT METHOD*

STEPHEN G. NASH[†] AND ARIELA SOFER[†]

Abstract. The theory of self-concordance in convex optimization has been used to analyze the complexity of interior-point methods based on Newton's method. For large problems, it may be impractical to use Newton's method; here we analyze a truncated-Newton method, in which an approximation to the Newton search direction is used. In addition, practical interior-point methods often include enhancements such as extrapolation that are absent from the theoretical algorithms analyzed previously. We derive theoretical results that apply to such an algorithm, one similar to a sophisticated computer implementation of a barrier method. The results for a single barrier subproblem are a satisfying extension of the results for Newton's method. When extrapolation is used in the overall barrier method, however, our results are more limited. We indicate (by both theoretical arguments and examples) why more elaborate results may be difficult to obtain.

Key words. large-scale optimization, complexity, convex programming, self-concordance, interior-point method, truncated-Newton method

AMS subject classifications. 49M30, 49M37, 49M45, 65K05, 90C06, 90C25, 90C30

PII. S1052623496306620

1. Introduction. In their 1993 book [16], Nesterov and Nemirovskii derive complexity results for convex optimization problems. Their basic algorithm is an interior-point method where each subproblem is solved using a damped Newton method. If a nonlinear optimization problem is large (and hence complexity is an important issue), then Newton's method is not normally used because of its computational costs, so these results might be considered primarily of theoretical interest. Our goal in this paper is to derive comparable complexity results for algorithms that more closely resemble practical interior-point algorithms for large-scale optimization (see, e.g., [1, 9, 10, 11, 13, 19, 23]).

The interior-point method we analyze is strongly related to the barrier method in [13]. The essential features of this algorithm are that each barrier subproblem is solved approximately using a truncated-Newton method. Then the solutions to the subproblems are extrapolated to obtain an initial guess for a new subproblem. Many of the enhancements discussed in [13]—such as preconditioning, a specialized matrix-vector product, and a numerically stable formula for the search direction—fit into the theoretical framework used here. The major exception is the line search (see below).

We derive a bound on the number of truncated-Newton iterations required to solve a barrier subproblem to within some tolerance. Each truncated-Newton iteration involves the approximate solution of the Newton equations via, say, the conjugate-gradient method, requiring at most $O(n^3)$ computations in exact arithmetic, although typically the number of computations would be $O(n)$ or $O(n^2)$ in problems where the Hessian matrix is sparse. In the algorithm analyzed here, a prescribed step length is used, so there is no line search. (Here is where the theoretical and practical algorithms differ, since a practical method would likely use an adaptive line search based on

*Received by the editors July 15, 1996; accepted for publication (in revised form) May 21, 1997; published electronically July 20, 1998. This research was supported by National Science Foundation grant DMI-9414355.

<http://www.siam.org/journals/siopt/8-3/30662.html>

[†]Operations Research and Engineering Department, George Mason University, Fairfax, VA 22030 (snash@gmu.edu, asofer@gmu.edu).

minimizing a one-dimensional approximation to the barrier function.) Ignoring the computations for evaluating the gradient and Hessian, the algorithm determines the solution to within a tolerance $\epsilon = 2^{-M}$ in a number of operations that is polynomial in M and the problem dimensions. If polynomial algorithms exist to evaluate the gradient and Hessian, the overall algorithm for the barrier subproblem is polynomial in the dimensions of the optimization problem.

The theoretical result that we obtain reduces to the result for Newton's method if the inner convergence tolerance for the truncated-Newton method is set to zero. For this reason, we consider this result to be a satisfying extension of the theory for Newton's method.

In the second major part of the paper we analyze how a simple linear extrapolation scheme can accelerate the algorithm by providing improved initial guesses for each subproblem. We show that improved performance can be achieved by an algorithm based on linear extrapolation, when barrier subproblems are solved exactly. We also indicate, via an example, that it may be difficult to derive complexity results either when subproblems are solved inexactly, or when higher-order extrapolation is used.

Our analysis is based on the framework established in Nesterov and Nemirovskii (1993) and Nemirovskii (1994) as adapted by us in our book [14]. In the rest of the paper we cite theoretical results for an algorithm based on Newton's method. These results are due to Nesterov and Nemirovskii, although we frequently cite [14] because our discussion here more closely follows the organization and notation of that book. (Related discussions can be found in [3, 8, 22].)

In the barrier method, we assume that the barrier function is *self-concordant*, a property that we define below. Self-concordant barrier functions were introduced in [16]; recent work on this topic includes [4, 6, 7, 8, 17]. For linear programs and convex quadratic programs, the ordinary logarithmic barrier function can be used. Related barrier functions can be used for semidefinite programming [18, 20, 21, 24]. It is possible to prove that, for any convex feasible region with the properties we specify below, there exists an appropriate self-concordant barrier function, although in general it may not be practical for computation. Thus the results we describe here provide a general theoretical approach for solving convex programming problems.

Two major results are required to prove that the overall algorithm is polynomial. The first states that if the truncated-Newton method is applied to a single barrier subproblem, and the initial guess is "close" to the solution, then the number of iterations required to find an approximate solution of this subproblem is bounded. This is the topic of section 3. The second states that the linear extrapolation can improve the initial guess for the next subproblem. This topic is addressed in section 4.

These two results are less obvious than they might at first seem. If any constraint in the convex programming problem is binding at the solution, then the solution will be on the boundary of the feasible region where the barrier function has a singularity. Since standard convergence results for Newton-type methods assume that the Hessian at the solution has a bounded condition number, a traditional analysis is not appropriate.

To analyze the behavior of Newton-type methods in this case, we must in some manner take this singularity into account. To do this, we define a norm $\|\cdot\|_x$ in terms of the Hessian of the barrier function evaluated at a point x . We will measure "closeness" in terms of this norm.

This norm depends on the Hessian, and changes as the variables change. For this norm to be useful, the rate of change of the Hessian matrix must not be too great. This reasoning leads to the imposition of a bound on the third derivatives of the barrier

function in terms of the Hessian (see section 2). This bound is all that is required to prove the first major result corresponding to the behavior of the truncated-Newton method on a single barrier subproblem.

To prove that the approximate solution of one subproblem will not be too far from the solution of the next subproblem, it is necessary that the values of the barrier functions not change too quickly as the barrier parameter changes. To guarantee this, we impose a bound on the first derivatives of the barrier functions in terms of the Hessian (see section 4). By measuring all quantities in terms of the Hessian, we are able to circumvent the difficulties associated with the singularity of the barrier function at the solution.

If the barrier function has these properties, then an interior-point method can be designed so that the optimal solution of a convex programming problem can be found (to within some tolerance) using a polynomial number of truncated-Newton iterations.

Practical experience suggests that an improved barrier method can be obtained if the approximate solutions to the subproblems are extrapolated to produce an initial guess for the next subproblem. We analyze this idea in section 4. We are able to derive some theory to support this idea in the case when linear extrapolation is used, and when the subproblems are solved exactly. Through an example, we suggest that it may be difficult to obtain a comparable result for either a more elaborate algorithm (with higher-order extrapolation) or a more realistic algorithm (with the subproblems solved inexactly).

2. Basics. In this section, we define self-concordance and establish some basic lemmas. An extensive discussion of the theory of self-concordance is given in the book by Nesterov and Nemirovskii [16]. The presentation here parallels our book [14].

Let S be a bounded, closed, convex subset of $\mathbb{R}^n$ with nonempty interior $\text{int } S$. (The assumption that S is bounded is not that important, since we could modify the optimization problem by adding artificial, very large bounds on the variables.) Let F be a convex function defined on the set S , and assume that F has three continuous derivatives. Then F is *self-concordant* on S if

- (i) (barrier property) $F(x_i) \rightarrow \infty$ along every sequence $\{x_i\} \subset \text{int } S$ converging to a boundary point of S .
- (ii) (differential inequality) F satisfies

$$|\nabla^3 F(x)[h, h, h]| \leq 2(h^T \nabla^2 F(x) h)^{3/2}$$

for all $x \in \text{int } S$ and all $h \in \mathbb{R}^n$.

In this definition,

$$\nabla^3 F(x)[h_1, h_2, h_3] \equiv \frac{\partial^3}{\partial t_1 \partial t_2 \partial t_3} F(x + t_1 h_1 + t_2 h_2 + t_3 h_3) \Big|_{t_1=t_2=t_3=0},$$

that is, it is a third-order directional derivative of F .

As an example, the logarithmic barrier function

$$F(x) = - \sum_{i=1}^m \log(a_i^T x - b_i)$$

is self-concordant on the set $S = \{x : a_i^T x - b_i \geq 0, i = 1, \dots, m\}$.

The constant 2 in the definition is arbitrary. If instead

$$|\nabla^3 F(x)[h, h, h]| \leq C(h^T \nabla^2 F(x) h)^{3/2}$$

for some constant C , then the scaled function $\hat{F} = \frac{1}{4}C^2F$ is self-concordant. The number 2 is used in the definition so that the function $F(x) = -\log x$ is self-concordant without any scaling.

We make several assumptions to simplify our discussion. They are not essential; in fact, almost identical results can be proved without these assumptions. We assume that $\nabla^2 F(x)$ is nonsingular for all $x \in \text{int } S$. (See Theorem 2.1.1 in [16] for an approach that avoids this assumption.) This allows us to define a norm as

$$\|h\|_x \equiv (h^T \nabla^2 F(x) h)^{1/2}.$$

We also assume that F has a minimizer $x^* \in \text{int } S$. Because F is convex, these assumptions guarantee that x^* is the unique minimizer of F in S .

The following lemmas indicate some basic properties of self-concordant functions. The first shows that the third-order directional derivative can be bounded using this norm.

LEMMA 2.1. *If F is self-concordant on S , then*

$$|\nabla^3 F(x)[h_1, h_2, h_3]| \leq 2 \|h_1\|_x \|h_2\|_x \|h_3\|_x$$

for all $x \in \text{int } S$ and for all $h_1, h_2, h_3 \in \mathbb{R}^n$.

Proof. See [8] or [16]. $\square$

The next lemma bounds how rapidly a self-concordant function $F(x)$ and its Hessian can change if a step is taken whose norm is less than one. The first result is an analogue of a Taylor series expansion for a self-concordant function. The second is a bound on how rapidly the norm can change when x changes.

LEMMA 2.2. *Let F be self-concordant on S . Let $x \in \text{int } S$ and suppose that $\|h\|_x < 1$. Then $x + h \in \text{int } S$, and*

$$(2.1) \quad F(x) + \nabla F(x)^T h + \beta(-\|h\|_x) \leq F(x + h) \leq F(x) + \nabla F(x)^T h + \beta(\|h\|_x),$$

where

$$\beta(s) = -\log(1 - s) - s = \frac{s^2}{2} + \frac{s^3}{3} + \frac{s^4}{4} + \cdots.$$

The lower bound in (2.1) is satisfied even if $\|h\|_x \geq 1$. Furthermore, for any $g \in \mathbb{R}^n$,

$$(1 - \|h\|_x) \|g\|_x \leq \|g\|_{x+h} \leq (1 + \|h\|_x) \|g\|_x.$$

Proof. See [16]. $\square$

Our convergence results for a barrier subproblem are phrased in terms of a quantity called the *Newton decrement*. It is defined below. The Newton decrement measures the norm of the Newton direction, but indirectly it can be interpreted as a “proximity measure” for the distance to the barrier trajectory. We use the Newton decrement in place of more traditional measures of convergence, such as $\|x - x^*\|$ and $|F(x) - F(x^*)|$.

If $x \in \text{int } S$ and p_N is the Newton direction for F at x , then the *Newton decrement* of F at x is

$$\delta = \delta(F, x) = \|p_N\|_x.$$

Consider the Taylor series approximation to $F(x + h)$ given by

$$F(x) + \nabla F(x)^T h + \frac{1}{2} h^T \nabla^2 F(x) h = F(x) + \nabla F(x)^T h + \frac{1}{2} \|h\|_x^2.$$

The Newton direction p_N minimizes this approximation and is the solution to

$$\nabla^2 F(x)p_N = -\nabla F(x).$$

The optimal value of the Taylor series approximation is

$$F(x) - \frac{1}{2}\delta(F, x)^2,$$

indicating why $\delta(F, x)$ is called the Newton decrement.

We have the following lemma.

LEMMA 2.3. *The Newton decrement satisfies*

$$\delta(F, x) = \max \{ \nabla F(x)^T h : \|h\|_x \leq 1 \}.$$

Proof. See [16]. $\square$

We will obtain bounds on $F(x) - F(x^*)$ and $\|x - x^*\|_{x^*}$ in terms of the Newton decrement, and we will also measure the progress at each iteration of the truncated-Newton method in terms of the Newton decrement. Thus, statements about the convergence of the method in terms of the Newton decrement will indirectly provide us with information about convergence as measured in the more traditional ways.

3. Convergence of a truncated-Newton method. We now study the consequences of using a truncated-Newton method, rather than Newton's method, to minimize a self-concordant function

$$\min_{x \in S} F(x),$$

where S and F are as in the previous section.

In the truncated-Newton method, a search direction p will be computed that satisfies the acceptance criterion

$$(3.1) \quad \|p - p_N\|_x \leq \epsilon \|p_N\|_x,$$

where x is the current estimate of the solution to the barrier subproblem, p_N is the Newton direction, and ϵ is some tolerance. For simplicity, we assume that the tolerance ϵ is fixed, although similar results could be derived in the case where ϵ varies from iteration to iteration.

It is not that important how the search direction is computed, as long as the number of arithmetic operations is polynomial in the dimensions of the problem. Practical truncated-Newton methods often use the conjugate-gradient method which, in exact arithmetic, is guaranteed to converge to the Newton direction in a finite number of operations. Thus, in exact arithmetic, this would be an appropriate procedure.

The acceptance criterion (3.1) is impractical since it involves the Newton direction p_N . It is, however, closely related to practical rules for terminating the inner iteration of a truncated-Newton method. If we define

$$Q(p) \equiv p^T \nabla F(x) + \frac{1}{2} p^T \nabla^2 F(x) p$$

to be the value of the quadratic model for $F(x)$ at p , then

$$\frac{\|p - p_N\|_x^2}{\|p_N\|_x^2} = \left| \frac{Q(p) - Q(p_N)}{Q(p_N)} \right|.$$

In [12] it is recommended that the inner iteration of a truncated-Newton method be terminated based on the value of the quadratic model, and the barrier method in [13] uses a related rule.

Other practical acceptance rules are based on the value of the relative residual, and have the form

$$\frac{\|\nabla^2 F(x)p + \nabla F(x)\|_2}{\|\nabla F(x)\|_2} \leq \eta$$

for some tolerance η [2]. It is straightforward to derive that

$$\frac{1}{\sqrt{\text{cond}_2(\nabla^2 F(x))}} \frac{\|p - p_N\|_x}{\|p_N\|_x} \leq \frac{\|\nabla^2 F(x)p + \nabla F(x)\|_2}{\|\nabla F(x)\|_2} \leq \sqrt{\text{cond}_2(\nabla^2 F(x))} \frac{\|p - p_N\|_x}{\|p_N\|_x},$$

where $\text{cond}_2(\nabla^2 F(x))$ is the condition number of $\nabla^2 F(x)$ in the 2-norm. These inequalities provide a further demonstration of the relationship of (3.1) with practical termination rules for the inner iteration of the truncated-Newton method.

Some useful consequences of the acceptance criterion (3.1) are stated in the following lemma.

LEMMA 3.1. *Suppose that the acceptance criterion (3.1) is satisfied at x . Then*

$$(3.2) \quad (1 - \epsilon) \|p_N\|_x \leq \|p\|_x \leq (1 + \epsilon) \|p_N\|_x$$

and

$$(3.3) \quad p^T \nabla F(x) \leq -\frac{1}{1 + \epsilon} \|p\|_x^2.$$

Proof. The first result is a straightforward consequence of (3.1). The second is obtained by squaring the acceptance criterion. Hence,

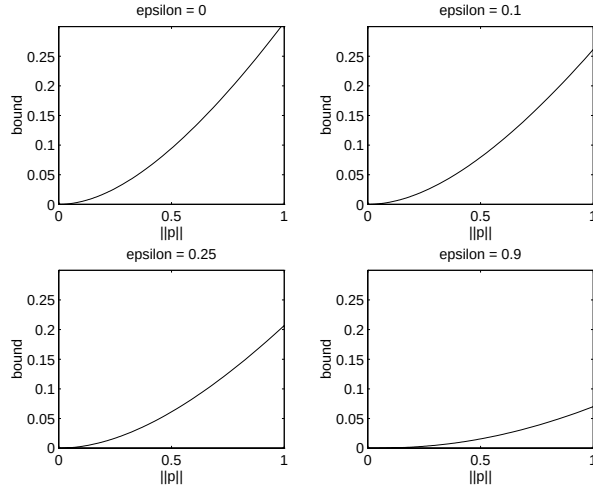
$$\begin{aligned} & (p - p_N)^T \nabla^2 F(x)(p - p_N) \leq \epsilon^2 p_N^T \nabla^2 F(x) p_N \\ \implies & p^T \nabla^2 F(x) p + p_N^T \nabla^2 F(x) p_N - 2p^T \nabla^2 F(x) p_N \leq \epsilon^2 p_N^T \nabla^2 F(x) p_N \\ \implies & (1 - \epsilon^2) \|p_N\|_x^2 + \|p\|_x^2 \leq 2p^T \nabla^2 F(x) p_N = -2p^T \nabla F(x) \\ \implies & \left(\frac{1 - \epsilon^2}{(1 + \epsilon)^2} + 1 \right) \|p\|_x^2 \leq -2p^T \nabla F(x) \\ \implies & p^T \nabla F(x) \leq -\frac{1}{1 + \epsilon} \|p\|_x^2. \quad \square \end{aligned}$$

The function F will be minimized using a “damped” truncated-Newton method, that is, a step is taken along the truncated-Newton direction but with a specified step length that is less than one. If we denote the search direction at x by p , then the method is defined by

$$x_+ = x + \frac{1}{1 + \|p\|_x} p.$$

The reason for including this step length is that the resulting displacement will always have norm less than one, so that Lemma 2.2 applies. It also guarantees that the damped truncated-Newton step is well defined, in the sense that the iterates remain in $\text{int } S$.

As the method converges and the truncated-Newton direction approaches zero, the step length approaches one, so that (asymptotically) rapid rates of convergence can be attained. The rest of this section develops the properties of the damped truncated-Newton method.

FIG. 3.1. Bound on $F(x) - F(x_+)$ for various values of ϵ .

The next lemma gives a lower bound on how much the function F will be decreased by a step of the damped truncated-Newton method.

LEMMA 3.2. *If x_+ is the result of the damped truncated-Newton iteration, then $x_+ \in \text{int } S$ and*

$$F(x) - F(x_+) \geq \frac{\|p\|_x}{1 + \|p\|_x} \left[\frac{\|p\|_x}{1 + \epsilon} + 1 \right] - \log(1 + \|p\|_x).$$

Proof. Let $\alpha = 1/(1 + \|p\|_x)$ be the step length. Using Lemma 2.2 and (3.3) we obtain

$$\begin{aligned} F(x_+) &\leq F(x) + \alpha \nabla F(x)^T p + \beta(\alpha \|p\|_x) \\ &\leq F(x) - \frac{1}{1 + \|p\|_x} \frac{\|p\|_x^2}{1 + \epsilon} - \log(1 - \alpha \|p\|_x) - \alpha \|p\|_x \\ &= F(x) - \frac{\|p\|_x}{1 + \|p\|_x} \left[\frac{\|p\|_x}{1 + \epsilon} + 1 \right] + \log(1 + \|p\|_x). \end{aligned}$$

The desired result is just a rearrangement of this last inequality. $\square$

The lemma provides a lower bound for $F(x) - F(x_+)$. The lower bound is zero when $\|p\|_x = 0$, and is positive and strictly increasing for $\|p\|_x > 0$. (The derivative of the right-hand side with respect to $\|p\|_x$ is positive.) The result gives a lower bound on how much progress is made at each truncated-Newton iteration. Figure 3.1 illustrates this lower bound for various values of ϵ .

If $\|p\|_x$ remains large, then the truncated-Newton method must decrease the value of $F(x)$ by a nontrivial amount. Since the function is bounded below on S , this cannot go on indefinitely, and $\|p\|_x$ must ultimately become small.

The next theorem analyzes the convergence of the method in the case where $\delta(F, x)$ is small. Since $\delta(F, x)$ and $\|p\|_x$ are related by (3.2), these two results provide a bound on the number of truncated-Newton iterations required to solve the optimization problem to within some tolerance. (This argument is made precise in Theorem 3.4.) The theorem also determines a bound on $\|x - x^*\|_{x^*}$ in terms of $\delta(F, x)$, and thus shows that if $\delta(F, x)$ is small, then the norm of the error is small as well.

THEOREM 3.3. *If $x \in \text{int } S$, then*

$$\delta(F, x_+) \leq (1 + \epsilon)(2 + \epsilon)\delta(F, x)^2 + \epsilon\delta(F, x).$$

Let x^* be the minimizer of F in S , and assume that $\delta(F, x) < 1$. Then

$$\|x - x^*\|_{x^*} \leq \frac{\delta(F, x)}{1 - \delta(F, x)}.$$

Proof. The proof is in two parts, proving each of the results in turn.

Part 1: We will derive the bound on $\delta(F, x_+)$. Let p be the truncated-Newton direction at x , p_N be the Newton direction at x , and $\alpha = 1/(1 + \|p\|_x)$. Thus $x_+ = x + \alpha p$. For any $h \in \mathbb{R}^n$ we define

$$\psi(t) = \nabla F(x + tp)^T h.$$

This function is twice continuously differentiable for $0 \leq t \leq \alpha$, with

$$\begin{aligned} \psi'(t) &= p^T \nabla^2 F(x + tp) h, \\ \psi''(t) &= \nabla^3 F(x + tp)[h, p, p]. \end{aligned}$$

By Lemma 2.1 and Lemma 2.2 we have

$$\begin{aligned} |\psi''(t)| &\leq 2 \|h\|_{x+tp} \|p\|_{x+tp}^2 \\ &\leq 2(1 - t \|p\|_x)^{-1} \|h\|_x (1 - t \|p\|_x)^{-2} \|p\|_x^2 \\ &= 2(1 - t \|p\|_x)^{-3} \|h\|_x \|p\|_x^2. \end{aligned}$$

Note that

$$\int_0^\alpha \left[\int_0^t 2(1 - \tau \|p\|_x)^{-3} \|p\|_x^2 d\tau \right] dt = \frac{\alpha^2 \|p\|_x^2}{1 - \alpha \|p\|_x}.$$

Then

$$\begin{aligned} \nabla F(x_+)^T h &= \psi(\alpha) \leq \psi(0) + \alpha \psi'(0) + \|h\|_x \int_0^\alpha \left[\int_0^t 2(1 - \tau \|p\|_x)^{-3} \|p\|_x^2 d\tau \right] dt \\ &= \nabla F(x)^T h + \alpha p^T \nabla^2 F(x) h + \frac{\alpha^2 \|h\|_x \|p\|_x^2}{1 - \alpha \|p\|_x} \\ &= \nabla F(x)^T h + \alpha p_N^T \nabla^2 F(x) h + \alpha(p - p_N)^T \nabla^2 F(x) h + \frac{\|p\|_x^2 \|h\|_x}{1 + \|p\|_x} \\ &= (1 - \alpha) \nabla F(x)^T h + \alpha(p - p_N)^T \nabla^2 F(x) h + \frac{\|p\|_x^2 \|h\|_x}{1 + \|p\|_x} \\ &\leq \frac{\|p\|_x}{1 + \|p\|_x} \nabla F(x)^T h + \frac{1}{1 + \|p\|_x} (p - p_N)^T \nabla^2 F(x) h \\ &\quad + \frac{(1 + \epsilon)^2 \|h\|_x \|p_N\|_x^2}{1 + \|p\|_x} \\ &\leq \frac{(1 + \epsilon) \|p_N\|_x^2 \|h\|_x}{1 + \|p\|_x} + \frac{(1 + \epsilon)^2 \|h\|_x \|p_N\|_x^2}{1 + \|p\|_x} \\ &\quad + \frac{1}{1 + \|p\|_x} \|p - p_N\|_x \|h\|_x \\ &\leq \frac{1}{1 + \|p\|_x} \{[(1 + \epsilon) + (1 + \epsilon)^2] \|p_N\|_x^2 + \epsilon \|p_N\|_x\} \frac{1}{1 - \alpha \|p\|_x} \|h\|_{x_+} \\ &= \|h\|_{x_+} \{(1 + \epsilon)(2 + \epsilon) \|p_N\|_x^2 + \epsilon \|p_N\|_x\}. \end{aligned}$$

Since this is true for all h we have

$$\delta(F, x_+) = \max\{\nabla F(x_+)^T h : \|h\|_{x_+} \leq 1\} \leq (1 + \epsilon)(2 + \epsilon)\delta(F, x)^2 + \epsilon\delta(F, x).$$

This completes Part 1 of the proof.

Part 2: The proof of $\|x - x^*\|_{x^*} \leq (\delta(F, x)/1 - \delta(F, x))$ can be found in [14] or [15]. $\square$

We conclude with a summary theorem. It provides a bound on the number of truncated-Newton iterations required to minimize F to within a tolerance.

THEOREM 3.4. *Let S be a bounded, closed, convex subset of $\mathbb{R}^n$ with nonempty interior, and let $F(x)$ be a convex function that is self-concordant on S . Given an initial guess $x_0 \in \text{int } S$, the damped truncated-Newton method is defined by the recurrence*

$$x_{i+1} = x_i + \frac{1}{1 + \|p\|_{x_i}} p,$$

where p is an approximation to the Newton direction p_N at x_i satisfying

$$\|p - p_N\|_x \leq \epsilon \|p_N\|_x$$

for some tolerance $0 \leq \epsilon < 1$. Then $x_i \in \text{int } S$ for all i , and

$$F(x_{i+1}) \leq F(x_i) - \left[\frac{\|p\|_{x_i}}{1 + \|p\|_{x_i}} \left(\frac{\|p\|_{x_i}}{1 + \epsilon} + 1 \right) - \log(1 + \|p\|_{x_i}) \right].$$

At some iteration i we must have

$$\delta(F, x_i) \leq \bar{\delta} \equiv \frac{(1 - \epsilon)}{2(1 + \epsilon)(2 + \epsilon)}$$

and then for every $j \geq i$ we have

$$\begin{aligned} \delta(F, x_{j+1}) &\leq (1 + \epsilon)(2 + \epsilon)\delta(F, x_j)^2 + \epsilon\delta(F, x_j) \leq \frac{1}{2}(1 + \epsilon)\delta(F, x_j) < \delta(F, x_j), \\ F(x_j) - F(x^*) &\leq \beta(\delta(F, x_j)) \leq \frac{\delta(F, x_j)^2}{2(1 - \delta(F, x_j))}, \\ \|x_j - x^*\|_{x^*} &\leq \frac{\delta(F, x_j)}{1 - \delta(F, x_j)}. \end{aligned}$$

The number of truncated-Newton steps required to find a point $x \in S$ with $\delta(F, x) \leq \kappa < 1$ is bounded by

$$C ([F(x_0) - F(x^*)] + \log(1/\kappa))$$

for some constant C that depends on ϵ and that decreases as $\epsilon \rightarrow 0$. If $\epsilon = 0$, then the number of steps is bounded by

$$\hat{C} ([F(x_0) - F(x^*)] + \log \log(1/\kappa))$$

for some constant $\hat{C}$.

Proof. The bound on $F(x_j) - F(x^*)$ is derived in [16]. The remaining conclusions are consequences of the earlier results. The formulae for the constants C and $\hat{C}$ can be derived as follows. Let $\delta_j = \delta(F, x_j)$. Let i be the first iteration for which $\delta_i \leq \bar{\delta}$. Such an index i must exist, since F is bounded below on S . If $j < i$ and hence $\delta_j > \bar{\delta}$, then by (3.2)

$$\|p\|_{x_j} \geq \omega \equiv (1 - \epsilon)\bar{\delta}.$$

It follows from Lemma 3.2 that

$$F(x_j) - F(x_{j+1}) \geq \left[\frac{\omega}{1+\omega} \left(\frac{\omega}{1+\epsilon} + 1 \right) - \log(1+\omega) \right] \equiv c_0.$$

Thus the number of initial iterations i is at most

$$\left\lceil \frac{F(x_0) - F(x^*)}{c_0} \right\rceil.$$

The progress of the later iterations is described by Theorem 3.3. If $j \geq i$, then

$$\delta_{j+1} \leq \frac{1}{2}(1+\epsilon)\delta(F, x_j) \leq \left[\left(\frac{1}{2} \right) (1+\epsilon) \right]^{j-i} \bar{\delta} \leq \frac{1}{4} \left[\left(\frac{1}{2} \right) (1+\epsilon) \right]^{j-i}.$$

Hence the number of later iterations is at most

$$\left\lceil \frac{\log(1/4\kappa)}{\log(2/(1+\epsilon))} \right\rceil.$$

Summing these two bounds determines the constant C .

If $\epsilon = 0$ then at the later iterations

$$\delta_{j+1} \leq 2\delta_j^2,$$

and a similar analysis determines $\hat{C}$. $\square$

If $\epsilon = 0$, and the truncated-Newton method computes the Newton direction, then the theorem is the same as that reported for Newton's method in [14, 15]. It establishes the polynomiality of the truncated-Newton method applied to a barrier subproblem (see section 1 for details).

4. Extrapolation. In section 3 we analyzed the behavior of a truncated-Newton method when applied to a single barrier subproblem. We now consider the overall interior-point method based on solving a sequence of subproblems. Our main concern is with the effects of extrapolating the (approximate) solutions of several subproblems to obtain an improved initial guess for the next subproblem.

The complexity results for the overall method depend on an additional assumption, that is, a bound on the first derivative of the barrier function. Although we do not make much direct use of this assumption in this paper, it underlies many of our comments, and so we state it here.

Let S be a set with the same properties as in section 3. Following [16], a self-concordant function F on S is a *self-concordant barrier function* for S if, for some constant $\nu > 0$,

$$(4.1) \quad |\nabla F(x)^T h| \leq \nu^{1/2} \|h\|_x$$

for all $x \in \text{int } S$ and all $h \in \mathbb{R}^n$. We may assume, without loss of generality, that $\nu \geq 1$.

The function $F(x) = -\sum_{i=1}^n \log(x_i)$ is a self-concordant barrier function with $\nu = n$ for $S = \{x : x \geq 0\}$; the function

$$F(x) = -\sum_{i=1}^m \log(a_i^T x - b_i)$$

is a self-concordant barrier function with $\nu = m$ for the set $S = \{x : a_i^T x \geq b_i, i = 1, \dots, m\}$. A self-concordant barrier function exists for any closed convex set S [16]; evaluating such a barrier function may not be computationally practical, however.

We also assume that the convex program is written in the following *standard form*:

$$(4.2) \quad \begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x \in S, \end{array}$$

where $c \neq 0$. An optimization problem with a general nonlinear objective function can be converted to this form by adding an additional variable and constraint.

The problem (4.2) will be solved using a path-following method of the following form. For $\rho > 0$ we define

$$F_\rho(x) = \rho c^T x + F(x),$$

where F is a self-concordant barrier function for the set S with parameter $\nu \geq 1$, and where the Hessian of F is nonsingular for all $x \in \text{int } S$. (The nonsingularity assumption is not essential; see [16].) Let $x^*(\rho)$ be the minimizer of $F_\rho(x)$ for $x \in \text{int } S$. Our method will generate $x_i \approx x^*(\rho_i)$ for $\rho_i \rightarrow +\infty$.

A complexity result for a particular algorithm based on this approach and Theorem 3.4 can be proved as in [14, 15]. In this algorithm, $x_i \in \text{int } S$ is accepted as an approximate minimizer of F_{ρ_i} if

$$\delta(F_{\rho_i}, x_i) \leq \kappa$$

for some $0 < \kappa \leq \frac{1}{2}$, and then x_i is used as an initial guess for minimizing $F_{\rho_{i+1}}$.

THEOREM 4.1. *Suppose that we solve the problem (4.2) on a bounded, closed, convex domain S using the path-following method described above, where F is a self-concordant barrier function with parameter $\nu \geq 1$. Let $0 < \kappa \leq \frac{1}{2}$ be the parameter in the acceptance test, and assume that the penalty parameters are updated via*

$$\rho_{i+1} = \left(1 + \frac{\theta}{\sqrt{\nu}}\right) \rho_i,$$

where $\theta > 0$. Assume that the method is initialized with ρ_0 and $x_0 \in \text{int } S$ where x_0 satisfies the acceptance test for F_{ρ_0} . Then

$$c^T x_i - c^T x^* \leq \frac{2\nu}{\rho_0} \left(1 + \frac{\theta}{\sqrt{\nu}}\right)^{-i} \quad \text{for } i = 1, 2, \dots$$

If a truncated-Newton method based on (3.1) is used to minimize F_{ρ_i} to within proximity κ , then the number of truncated-Newton iterations required to find an approximate solution to subproblem i does not exceed a constant $N_{\kappa, \theta, \epsilon}$ depending only on κ , θ , and ϵ . In particular, the total number of truncated-Newton iterations required to find an x satisfying $c^T x - c^T x^* \leq \mu$ is bounded above by

$$C_{\kappa, \theta, \epsilon} \sqrt{\nu} \log \left(\frac{2\nu}{\rho_0 \mu} \right),$$

with constant $C_{\kappa, \theta, \epsilon}$ depending only on κ , θ , and ϵ .

It is possible that a better initial guess for a subproblem (and hence a better algorithm) can be obtained by extrapolation of previous solutions. The technique of extrapolation—initially proposed by Fiacco and McCormick [5]—approximates the barrier trajectory $x(\rho)$ by a polynomial of degree q . The coefficients of the polynomial are computed from the solutions of $q + 1$ subproblems, and are then used to predict the solution of the barrier subproblem for the new value of ρ . The results in [5] indicate that extrapolation is a powerful computational tool. Our own computational

TABLE 4.1

Effect of cubic extrapolation (*iter* = number of truncated-Newton iterations, *ng* = number of gradient evaluations). Based on [13] with elaborate barrier algorithm, test problem 51, 1000 variables.

ρ	Extrapolation		No extrapolation	
	iter	ng	iter	ng
6.8×10^{-6}	1	4	1	4
6.8×10^{-5}	5	38	5	38
6.5×10^{-4}	6	77	8	77
6.8×10^{-3}	6	42	6	42
6.8×10^{-2}	8	58	12	83
6.8×10^{-1}	11	75	25	242
6.8×10^0	7	39	16	178
6.8×10^1	5	26	28	277
6.8×10^2	1	4	4	28
6.8×10^3	1	4	2	13
Totals	51	367	107	982

experiments [13] also indicate that better initial guesses (and better overall performance) can be obtained by extrapolating the solutions of a sequence of subproblems. An example illustrating the usefulness of extrapolation is summarized in Table 4.1; for details, see [13]. In this case, the use of cubic extrapolation reduces the number of truncated-Newton iterations by a factor of 2.1, and the number of gradient evaluations by a factor of 2.7.

(Complexity results for a predictor-corrector method can be found in [15]; this predictor-corrector method is a form of extrapolation, but appears to be less practical for large nonlinear programs than the technique used here. It is not clear that this predictor-corrector approach can be extended effectively to a truncated-Newton method.)

Lemma 4.2 below gives theoretical support to these computational results. The technique of extrapolation may have limitations, however, as we shall discuss in the latter part of this section.

We first examine linear extrapolation and assume that $x_i = x^*(\rho_i)$. Suppose that $\rho_{i+1} = (1 + \gamma_i)\rho_i$ for all i , where $\gamma_i > 0$. Then linear extrapolation predicts that

$$x(\rho_{i+1}) \approx x_i + \frac{\gamma_i}{\gamma_{i-1}} \Delta_i,$$

where

$$\Delta_i \equiv (1 + \gamma_{i-1})(x_i - x_{i-1}).$$

More generally, Δ_i can be considered as a “search direction” along the barrier trajectory, as in Lemma 4.2 below. The result shows that linear extrapolation can be used to produce initial guesses that are at least as good as when extrapolation is not used. For simplicity, we choose $\gamma_i \equiv \gamma$ for all i , although it would be easy to extend the result to the case where the γ_i ’s are not constant. (The lemma uses our general assumption that $\nabla^2 F$ is positive definite. If $\nabla^2 F$ is only positive semidefinite, then the lemma is still true under the assumption that $c^T \Delta_i \neq 0$.)

LEMMA 4.2. Let $\rho_{i+1} = (1 + \gamma)\rho_i$ for some $\gamma > 0$, and assume that $x_i = x^*(\rho_i)$. Define the linear extrapolation direction as

$$\Delta_i \equiv (1 + \gamma)(x_i - x_{i-1}).$$

If

$$F_{i+1}(x) = \rho_{i+1} c^T x + F(x),$$

then

$$\nabla F_{i+1}(x_i)^T \Delta_i = \gamma \rho_i (c^T \Delta_i) < 0,$$

that is, Δ_i is a descent direction for F_{i+1} at x_i . We define

$$\alpha_+ = \frac{1}{\|\Delta_i\|_{x_i}} \cdot \frac{1}{1 - \frac{\|\Delta_i\|_{x_i}}{\gamma \rho_i c^T \Delta_i}}.$$

Then $\alpha_+ > 0$. If $\alpha \leq \min\{1, \alpha_+\}$ then $x_i + \alpha \Delta_i \in \text{int } S$ and

$$F_{i+1}(x_i + \alpha \Delta_i) < F_{i+1}(x_i),$$

with a more specific upper bound provided by (2.1).

Proof. We first prove that $c^T x_i < c^T x_{i-1}$. A Taylor series expansion gives

$$F(x_i) = F(x_{i-1}) + \nabla F(x_{i-1})^T (x_i - x_{i-1}) + \frac{1}{2} (x_i - x_{i-1})^T \nabla^2 F(\xi) (x_i - x_{i-1}).$$

Because $x_i \neq x_{i-1}$ and $\nabla^2 F$ is positive definite, the last term is positive. Since $\nabla F(x_{i-1}) = -\rho_{i-1} c$, we obtain

$$F(x_i) > F(x_{i-1}) - \rho_{i-1} c^T (x_i - x_{i-1}),$$

which implies that

$$F(x_{i-1}) + \rho_{i-1} c^T x_{i-1} < F(x_i) + \rho_{i-1} c^T x_i.$$

Similarly, by switching the roles of x_i and x_{i-1} , we obtain

$$F(x_i) + \rho_i c^T x_i < F(x_{i-1}) + \rho_i c^T x_{i-1}.$$

If we multiply the first of these inequalities by ρ_i , the second by ρ_{i-1} , and rearrange, then

$$(\rho_i - \rho_{i-1}) F(x_{i-1}) < (\rho_i - \rho_{i-1}) F(x_i).$$

Since $\rho_i - \rho_{i-1} > 0$, then $F(x_{i-1}) < F(x_i)$. Combining this with the above results gives $c^T x_i < c^T x_{i-1}$, as desired.

We now use this result to prove that Δ_i is a descent direction. Since $x_i = x^*(\rho_i)$,

$$\nabla F_i(x_i) = \rho_i c + \nabla F(x_i) = 0.$$

Using these results, we obtain

$$\begin{aligned} \nabla F_{i+1}(x_i)^T \Delta_i &= (1 + \gamma) [\rho_{i+1} c + \nabla F(x_i)]^T (x_i - x_{i-1}) \\ &= (1 + \gamma) [(\rho_{i+1} - \rho_i) c]^T (x_i - x_{i-1}) \\ &= (1 + \gamma) \gamma \rho_i c^T (x_i - x_{i-1}) \\ &= \gamma \rho_i (c^T \Delta_i) < 0. \end{aligned}$$

This completes the first part of the proof.

The second part of the proof relies on (2.1). Using the formulae above, the upper bound in (2.1) becomes

$$F_{i+1}(x_i + \alpha \Delta_i) \leq \rho_i (1 + \gamma) c^T x_i + F(x_i) + \alpha \gamma \rho_i c^T \Delta_i + \beta (\|\alpha \Delta_i\|_{x_i}) \equiv \phi(\alpha).$$

Note that $\phi(0) = F_{i+1}(x_i)$ and that $\phi'(0) < 0$ by the assumptions in the lemma.

If we solve $\phi'(\alpha) = 0$, we obtain the solution

$$\alpha_+ = \frac{1}{\|\Delta_i\|_{x_i}} \cdot \frac{1}{1 - \frac{\|\Delta_i\|_{x_i}}{\gamma\rho_i c^T \Delta_i}}.$$

It is straightforward to verify that $\phi''(\alpha_+) > 0$, so that this is a local minimizer of ϕ . Since $c^T \Delta_i = (1 + \gamma)c^T(x_i - x_{i-1}) < 0$, we obtain that

$$\alpha_+ < \frac{1}{\|\Delta_i\|_{x_i}} \alpha_+ > 0.$$

If $\alpha \leq \min\{1, \alpha_+\}$, then

$$\|\alpha\Delta_i\|_{x_i} \leq \|\alpha_+\Delta_i\|_{x_i} < \frac{1}{\|\Delta_i\|_{x_i}} \|\Delta_i\|_{x_i} = 1,$$

and so $x_i + \alpha\Delta_i \in \text{int } S$.

Since the upper bound on $F_{i+1}(x_i + \alpha\Delta_i)$ is decreasing at $\alpha = 0$, and since $\alpha_+ > 0$ is a local minimizer of the upper bound, this completes the proof. $\square$

This result could be extended to the case where the subproblems are not solved exactly, as long as the magnitudes of $\|x_j - x^*(\rho_j)\|_{x_j}$ were sufficiently small so as not to interfere with the inequalities in the lemma. It appears to be difficult to generalize the above result greatly, however, as the discussion below indicates.

Lemma 4.2 shows that an appropriate step along the extrapolation direction can produce a decrease in the objective value of the barrier function for the updated barrier parameter. However, taking the “full” extrapolation step $(\gamma_i/\gamma_{i-1})\Delta_i$ is not guaranteed to be beneficial, no matter how slight the change in the barrier parameter. This is true even when the subproblems are solved exactly. To see this, let

$$\rho(\gamma) = (1 + \gamma)\rho_i,$$

and consider the extrapolated point

$$x(\gamma) = x_i + \frac{\gamma}{\gamma_{i-1}}(1 + \gamma_{i-1})(x_i - x_{i-1}).$$

The effect of extrapolation is described by

$$\psi(\gamma) \equiv F_{\rho(\gamma)}(x(\gamma)) - F_{\rho(\gamma)}(x_i) = (1 + \gamma)\rho_i c^T(x(\gamma) - x_i) + F(x(\gamma)) - F(x_i).$$

Now

$$\begin{aligned} \frac{d}{d\gamma}\psi(\gamma) &= \rho_i c^T(x(\gamma) - x_i) + \frac{(1 + \gamma)\rho_i}{\gamma_{i-1}}(1 + \gamma_{i-1})(c^T x_i - c^T x_{i-1}) \\ &\quad + \frac{1 + \gamma_{i-1}}{\gamma_{i-1}} \nabla F(x(\gamma))^T(x_i - x_{i-1}). \end{aligned}$$

Since $0 = \nabla F_{\rho_i}(x_i) = \rho_i c + \nabla F(x_i)$, we obtain

$$\begin{aligned} \frac{d}{d\gamma}\psi(0) &= \rho_i c^T(x_i - x_i) + \rho_i \frac{1 + \gamma_{i-1}}{\gamma_{i-1}}(c^T x_i - c^T x_{i-1}) - \rho_i \frac{1 + \gamma_{i-1}}{\gamma_{i-1}} c^T(x_i - x_{i-1}) \\ &= 0. \end{aligned}$$

Thus, for a small change in the barrier parameter (i.e., as $\gamma \rightarrow 0^+$), to first order there is no improvement in the objective value of the new barrier function at the extrapolated point.

Somewhat more insight can be obtained by analyzing $\psi''(0)$:

$$\psi''(0) = 2\rho_i \frac{1 + \gamma_{i-1}}{\gamma_{i-1}} c^T(x_i - x_{i-1}) + \left(\frac{1 + \gamma_{i-1}}{\gamma_{i-1}} \right)^2 (x_i - x_{i-1})^T \nabla^2 F(x_i) (x_i - x_{i-1}).$$

The second term is positive by convexity, but the first term is less than or equal to zero (see Lemma 4.2). If we define

$$v \equiv \frac{1 + \gamma_{i-1}}{\gamma_{i-1}} (x_i - x_{i-1}),$$

then we can write

$$\psi''(0) = \|v\|_{x_i}^2 - 2\nabla F(x_i)^T v.$$

From (4.1) and our comments above we obtain

$$\|v\|_{x_i}^2 \geq \psi''(0) \geq \|v\|_{x_i}^2 - 2\sqrt{\nu} \|v\|_{x_i}.$$

If $\|v\|_{x_i} > 2\sqrt{\nu}$, then the lower bound on $\psi''(0)$ is positive, and hence no improvement in the objective value is obtained. Thus the full extrapolation step is not guaranteed to be useful.

So far, we have only considered linear extrapolation. We now indicate via an example that higher-order extrapolation may lead to predicted solutions that are far worse than those obtained without extrapolation. In this example, we consider both exact and approximate solutions to subproblems. As above, an approximate solution x_i to a barrier subproblem will be accepted if $x_i \in \text{int } S$ and if

$$\delta(F_{\rho_i}, x_i) \leq \kappa$$

for some $\kappa \geq 0$.

Consider the one-variable problem

$$\begin{aligned} & \text{minimize} && x \\ & \text{subject to} && x \geq 0. \end{aligned}$$

This problem is already in standard form, and $\log(x)$ can be used as a barrier function with $\nu = 1$ in (4.1).

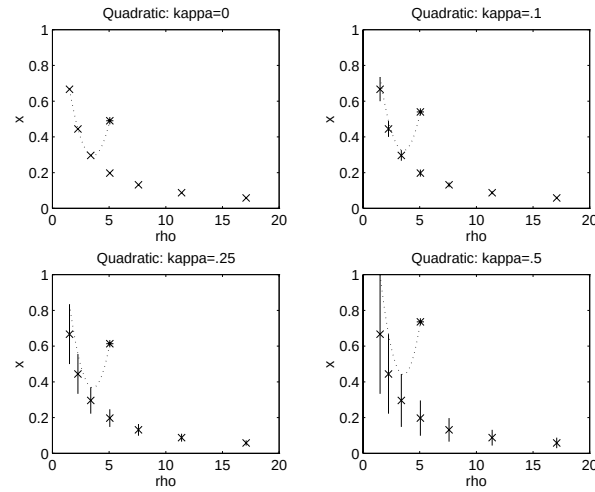
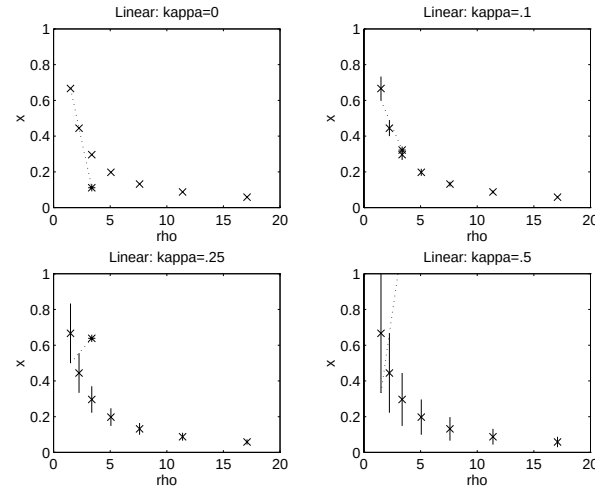
It is easy to analyze this example. The solution to the optimization problem is $x^* = 0$. The solution to the barrier subproblem is $x^*(\rho) = 1/\rho$. The norm has the value $\|h\|_x = |h/x|$. The Newton decrement is $\delta = \|p_N\|_x = |1 - x\rho|$. The solution to a subproblem is accepted if

$$\delta \leq \kappa \iff \rho^{-1}(1 - \kappa) \leq x \leq \rho^{-1}(1 + \kappa).$$

The penalty parameter is updated via $\rho_{i+1} = (1 + \gamma)\rho_i$.

Here we use $\gamma = \frac{1}{2}$ and consider four values of κ : $\kappa = 0, 0.1, 0.25, 0.5$. The choice $\kappa = 0$ corresponds to solving the subproblems exactly. We initialize the penalty parameter with $\rho_1 = 1$. In each case, the approximate solution to the subproblem is chosen at the upper bound of the acceptable range. This choice is consistent with the class of theoretical algorithms that we study in this paper, and is plausible for a practical algorithm.

In Figure 4.1 we show the results of applying quadratic extrapolation on this problem. The exact solutions of the first seven barrier subproblems are marked with $\times$. The vertical bars indicate the range of x values that satisfy the acceptance criterion for a subproblem. The dotted lines show the path of extrapolated approximate solutions, with a $*$ used to indicate the extrapolated initial guess.

FIG. 4.1. *Quadratic extrapolation of approximate solutions.*FIG. 4.2. *Linear extrapolation of approximate solutions.*

In each of the four cases, the extrapolated values are exceedingly poor initial guesses for the next subproblem. In fact, the extrapolation paths move away from, rather than toward, the solution of the next subproblem.

In Figure 4.2 we show the results of applying linear extrapolation. In this case, the approximate solutions are chosen as the lower bound of acceptable values for the first subproblem, and the upper bound for the second subproblem. This is allowed by the theory, but we think it unlikely that a practical algorithm could produce such approximate solutions.

In this case, even with linear extrapolation, the extrapolated points can (when κ is sufficiently large) point away from the solution of the next subproblem. Hence extrapolation is worse than doing nothing. As mentioned, we do not expect these circumstances to arise for a practical algorithm. Nevertheless, any complexity theory for such an algorithm would have to rule out this possibility, and hence would have to be based on a more elaborate theoretical framework than that used here.

This example suggests that an algorithm that uses extrapolation must monitor the effectiveness of the extrapolation scheme, and must not use it blindly. It also

suggests that the algorithm would have to be carefully designed so that inaccuracies in the solutions of barrier subproblems did not interfere with the performance of the extrapolation scheme.

We have not been able to derive complexity results for a more elaborate algorithm of this type. This example leads us to think that this would be a difficult enterprise.

REFERENCES

- [1] M. G. BREITFELD AND D. SHANNO, *Computational experience with penalty/barrier methods for nonlinear programming*, Ann. Oper. Res., 62 (1996), pp. 439–464.
- [2] R. S. DEMBO AND T. STEihaug, *Truncated-Newton algorithms for large-scale unconstrained optimization*, Math. Programming, 26 (1983), pp. 190–212.
- [3] D. DEN HERTOOG, *Interior Point Approach to Linear, Quadratic and Convex Programming: Algorithms and Complexity*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994.
- [4] D. DEN HERTOOG, F. JARRE, C. ROOS, AND T. TERLAKY, *A sufficient condition for self-concordance, with application to some classes of structured convex programming problems*, Math. Programming, 69 (1995), pp. 75–88.
- [5] A. V. FIACCO AND G. P. MCCORMICK, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley, Philadelphia, 1968.
- [6] O. GÜLER, *On the self-concordance of the universal barrier function*, SIAM J. Optim., 7 (1997), pp. 295–303.
- [7] S. HERZL AND M. J. TODD, *Two interior-point algorithms for a class of convex programming problems*, Optimization Methods and Software, 5 (1995), pp. 27–55.
- [8] F. JARRE, *Interior Point Methods via Self-Concordance or Relative Lipschitz Condition*, Habilitationsschrift, Bayerischen Julius-Maximilians-Universität Würzburg, 1994.
- [9] F. JARRE AND M. A. SAUNDERS, *A practical interior-point method for convex programming*, SIAM J. Optim., 5 (1995), pp. 149–171.
- [10] C. KANZOW, *An unconstrained optimization technique for large-scale linearly constrained minimization problems*, Computing, 53 (1994), pp. 101–117.
- [11] L. S. LASDON, J. PLUMMER AND G. YU, *Primal-dual and primal interior point algorithms for general nonlinear programs*, ORSA J. Comput., 7 (1995), pp. 321–332.
- [12] S. G. NASH AND A. SOFER, *Assessing a search direction within a truncated-Newton method*, Oper. Res. Lett., 9 (1990), pp. 219–221.
- [13] S. G. NASH AND A. SOFER, *A barrier method for large-scale constrained optimization*, ORSA J. Comput., 5 (1993), pp. 40–53.
- [14] S. G. NASH AND A. SOFER, *Linear and Nonlinear Programming*, McGraw-Hill, New York, 1996.
- [15] A. S. NEMIROVSKII, *Interior Point Polynomial Time Methods in Convex Programming*, Lecture notes, Faculty of Industrial Engineering and Management, Technion—The Israel Institute of Technology, 1994.
- [16] Y. NESTEROV AND A. S. NEMIROVSKII, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, 1994.
- [17] Y. E. NESTEROV AND M. J. TODD, *Primal-dual interior point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.
- [18] F. POTRA AND R. SHENG, *Homogeneous Interior-Point Algorithms for Semidefinite Programming*, Reports on Computational Mathematics 82, Department of Mathematics, The University of Iowa, Iowa City, IA, 1995.
- [19] R. SAIGAL, *An infinitely summable series implementation of interior-point methods*, Math. Oper. Res., 20 (1998), pp. 597–616.
- [20] J. F. STURM AND S. ZHANG, *On the Long Step Path-Following Method for Semidefinite Programming*, Report 9638/A, Econometric Institute, Erasmus University, Rotterdam, The Netherlands, 1996.
- [21] L. VANDENBERGHE AND S. BOYD, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.
- [22] S. A. VAVASIS, *Nonlinear Optimization: Complexity Issues*, Oxford University Press, New York, 1991.
- [23] M. H. WRIGHT, *Interior methods for constrained optimization*, in Acta Numerica 1992, A. Iserles, ed., Cambridge University Press, New York, 1992, pp. 341–407.
- [24] Y. ZHANG, *On extending some primal-dual interior point algorithms from linear programming to semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 365–386.

REGULARITY PROPERTIES OF A SEMISMOOTH REFORMULATION OF VARIATIONAL INEQUALITIES*

FRANCISCO FACCHINEI[†], ANDREAS FISCHER[‡], AND CHRISTIAN KANZOW[§]

Abstract. Variational inequalities over sets defined by systems of equalities and inequalities are considered. A new reformulation of the KKT conditions for the variational inequality as a system of equations is proposed. A related unconstrained minimization reformulation is also investigated. As a by-product of the analysis, a new characterization of strong regularity of KKT points is given.

Key words. variational inequality problem, KKT conditions, strong regularity

AMS subject classifications. 90C33, 65K05, 90C31

PII. S1052623496298194

1. Introduction. We consider the variational inequality problem $\text{VIP}(X, F)$, which is to find a vector $x^* \in X$ such that

$$(1.1) \quad F(x^*)^T(x - x^*) \geq 0 \quad \text{for all } x \in X,$$

where $X \subseteq \mathbb{R}^n$ is a closed set and $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is any given function which we assume to be at least continuously differentiable.

VIPs include as special cases complementarity problems and necessary optimality conditions for constrained optimization problems, and have many applications in the engineering and economic sciences; see, e.g., [13, 22, 23]. This has motivated an increasing interest in variational inequalities, from both the theoretical and the algorithmic points of view. One of the main paths followed in the theoretical study of variational inequalities has been their reduction to nonsmooth systems of (generalized) equations, which opens the way to the application of some sophisticated machinery developed in the latter field. In particular, the reformulation as a generalized equation has led to a dramatic improvement in our understanding of the properties of variational inequalities; see, for example, [19, 27, 28]. Another important kind of reformulation, which attracted much attention and also gave important insights into VIPs, is the one based on the normal map; see [7, 20, 31, 32] and references therein.

Naturally associated with the equation reformulation is the optimization approach, where the square of the norm of the system of equations is minimized in order to numerically find a solution of the variational inequality. Also in this case the use of the normal map equation reformulation has led to the development of robust and effective algorithms; see [2, 6, 16, 26, 33, 34].

In this paper, we propose a new equation reformulation of the KKT conditions of $\text{VIP}(X, F)$ (see below) and study some relevant properties of this reformulation and of the associated optimization problem. In order to explain this approach, a few words on the KKT system associated with $\text{VIP}(X, F)$ are in order.

*Received by the editors January 29, 1996; accepted (in revised form) June 4, 1997; published electronically July 21, 1998. This research was partially supported by NATO grant CRG 960137.

<http://www.siam.org/journals/siopt/8-3/29819.html>

[†]Dipartimento di Informatica e Sistemistica, Università di Roma “La Sapienza,” Via Buonarroti 12, I-00185 Roma, Italy (soler@dis.uniroma1.it).

[‡]Department of Mathematics, University of Dortmund, 44221 Dortmund, Germany (fischer@math.uni-dortmund.de).

[§]Institute of Applied Mathematics, University of Hamburg, Bundesstrasse 55, D-20146 Hamburg, Germany (kanzow@math.uni-hamburg.de).

We assume that the feasible set X can be represented as

$$(1.2) \quad X := \{x \in \mathbb{R}^n \mid h(x) = 0, g(x) \geq 0\},$$

where $h : \mathbb{R}^n \rightarrow \mathbb{R}^p$ and $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ are continuously differentiable functions. Then the following equations and inequalities are called the *KKT conditions* of problem $\text{VIP}(X, F)$:

$$(1.3) \quad \begin{aligned} F(x) + \nabla h(x)y - \nabla g(x)z &= 0, \\ h(x) &= 0, \\ g(x) \geq 0, z \geq 0, z^T g(x) &= 0. \end{aligned}$$

A triple $(x, y, z) \in \mathbb{R}^{n+p+m}$ satisfying (1.3) is called a *KKT point*.

There is a strong relationship between the KKT conditions (1.3) and the VIP (1.1). If $x^* \in \mathbb{R}^n$ is a (local) solution of $\text{VIP}(X, F)$ and if a constraint qualification holds (e.g., the Mangasarian–Fromovitz condition), then multiplier vectors $y^* \in \mathbb{R}^p$ and $z^* \in \mathbb{R}^m$ exist such that the vector $w^* := (x^*, y^*, z^*) \in \mathbb{R}^{n+p+m}$ is a KKT point of $\text{VIP}(X, F)$. Conversely, if all component functions h_i ($i = 1, \dots, p$) are affine and all component functions g_i ($i = 1, \dots, m$) are concave (so that X is a convex set), then the x -part of every KKT point $w^* = (x^*, y^*, z^*)$ is a solution of $\text{VIP}(X, F)$; see [17]. Note that F is not required to be monotone for this relationship to hold. Moreover, if X is a polyhedral set, then each solution of $\text{VIP}(X, F)$ corresponds to a KKT point and vice versa.

In order to reformulate the KKT conditions of $\text{VIP}(X, F)$ as a system of equations, we use the function $\varphi : \mathbb{R}^2 \rightarrow \mathbb{R}$ defined by

$$\varphi(a, b) := \sqrt{a^2 + b^2} - a - b.$$

This function was first used by Fischer [14] in order to reformulate the KKT conditions of an inequality constrained optimization problem. Since then it has become quite popular in the fields of linear and nonlinear complementarity, constrained optimization, and VIPs; see, e.g., the survey paper [15]. The main property of this function is the following characterization of its zeros:

$$\varphi(a, b) = 0 \iff a \geq 0, b \geq 0, ab = 0.$$

Therefore, the KKT conditions (1.3) can equivalently be written as the nonlinear system of equations

$$(1.4) \quad \Phi(w) := \Phi(x, y, z) = 0,$$

where $\Phi : \mathbb{R}^{n+p+m} \rightarrow \mathbb{R}^{n+p+m}$ is defined by

$$\Phi(w) = \Phi(x, y, z) := \begin{pmatrix} L(x, y, z) \\ h(x) \\ \phi(g(x), z) \end{pmatrix},$$

with

$$\begin{aligned} L(x, y, z) &:= F(x) + \nabla h(x)y - \nabla g(x)z, \\ \phi(g(x), z) &:= (\varphi(g_1(x), z_1), \dots, \varphi(g_m(x), z_m))^T \in \mathbb{R}^m. \end{aligned}$$

Associated with the nonlinear system of equations (1.4), we can consider the problem of minimizing its natural merit function, i.e.,

$$(1.5) \quad \Psi(w) \rightarrow \min,$$

where

$$\Psi(w) := \frac{1}{2} \Phi(w)^T \Phi(w) = \frac{1}{2} \|\Phi(w)\|^2.$$

Note that the function φ is not differentiable at the origin, so that the system (1.4) is a nonsmooth reformulation of the KKT conditions (1.3). However, it can be seen that Φ is a locally Lipschitz-continuous operator, while, remarkably, if h and g are C^2 -functions, Ψ is continuously differentiable; see Proposition 4.1 below.

Our main motivation for studying the reformulations (1.4) and (1.5) of the KKT conditions (1.3) comes from the field of algorithms for complementarity and optimization problems. There, it was recently shown that reformulations based on the function φ possess many valuable properties and allow the development of efficient solution methods [5, 8, 12]. Moreover, we think that the approach proposed in this paper also gives additional insight into the VIP that could fruitfully be exploited in the analysis of, e.g., stability and sensitivity properties. In particular we point out that we shall establish a new, appealing characterization of strongly regular KKT points of $\text{VIP}(X, F)$. We refer the reader to [5, 15] for a more detailed discussion on approaches based on the function φ and to [9, 10, 11] for applications of the results of this paper to the development of algorithms for the solution of problem $\text{VIP}(X, F)$.

In this paper we shall concentrate on two specific topics which, in the past few years, emerged as crucial points in reformulations of complementarity problems and VIPs as systems of equations or as optimization problems:

- (a) conditions which ensure the nonsingularity of all the generalized Jacobians of Φ (or at least ensure the existence of a nonsingular element in the generalized Jacobian), and
- (b) conditions which guarantee that a stationary point of Ψ is a solution of the variational inequality $\text{VIP}(X, F)$.

Points (a) and (b) are very relevant to the development of efficient algorithms [5, 9, 10], while point (a) could also pave the way to an application of the equation reformulation to the study of stability and sensitivity of solutions of variational inequalities.

The paper is organized as follows. In the next section we collect some basic definitions and results on coherent orientation. In section 3 we study conditions for the nonsingularity of the generalized Jacobians of Φ . In section 4 we are concerned with the important question of establishing conditions under which a stationary point of our merit function Ψ is a KKT point of $\text{VIP}(X, F)$. We present several conditions depending on the structure of the feasible set X .

Notation. A function $G : \mathbb{R}^n \rightarrow \mathbb{R}^t$ is called a C^k -function if it is k times continuously differentiable and an LC^k -function if it is a C^k -function whose k th derivative is locally Lipschitz-continuous. If $G : \mathbb{R}^n \rightarrow \mathbb{R}^t$ is a C^1 -function, we denote by $\nabla G(x)$ the gradient of G (i.e., the transpose of the Jacobian) at x . If G is locally Lipschitzian at x , $\partial G(x)$ indicates Clarke's [3] generalized Jacobian, a set of $t \times n$ matrices. When $t = 1$, $\partial G(x)$ is the generalized gradient and is usually regarded as a set of column vectors. For a matrix $A \in \mathbb{R}^{n \times t}$, $A = (a_{ij})$, and subsets $I \subseteq \{1, \dots, n\}$ and $J \subseteq \{1, \dots, t\}$, we denote by A_{IJ} the submatrix of A consisting of the elements a_{ij} , $i \in I$, $j \in J$. Similarly, given any vector $v \in \mathbb{R}^n$, $v = (v_1, \dots, v_n)^T$, and any subset

$I \subseteq \{1, \dots, n\}$, we denote by v_I the vector in $\mathbb{R}^{|I|}$ having the components $v_i, i \in I$. Let $A \in \mathbb{R}^{n \times n}$, $I \subseteq \{1, \dots, n\}$, and $\bar{I} := \{1, \dots, n\} \setminus I$ be the complementary subset of I . If we write

$$A = \begin{pmatrix} A_{II} & A_{I\bar{I}} \\ A_{\bar{I}I} & A_{\bar{I}\bar{I}} \end{pmatrix}$$

and assume that the principal submatrix A_{II} is nonsingular, then the Schur-complement of A_{II} in A is defined by $A/A_{II} := A_{\bar{I}\bar{I}} - A_{\bar{I}I}A_{II}^{-1}A_{I\bar{I}}$. A matrix $A \in \mathbb{R}^{n \times n}$ is a P_0 -matrix (P -matrix) if the determinant of each of its principal submatrices is nonnegative (positive). We recall that every positive semidefinite (positive definite) matrix is a P_0 -matrix (P -matrix), but the converse is not true in general. Finally, throughout the paper, $\|\cdot\|$ denotes the Euclidean vector norm.

2. Faces and coherently oriented matrices. Let B be a square t -dimensional matrix. The *orientation* of B is defined to be the sign of the determinant of B , and can therefore assume three possible values: $-1, 0$, and $+1$. Let L be a t -dimensional subspace of $\mathbb{R}^n$ and Q be a matrix whose columns form a basis for L . If A is an $n \times n$ matrix, then the matrix $Q^T A Q$ is called the *section* of A in the subspace L . The orientation of A on the subspace L is defined to be the orientation of $Q^T A Q$. It is easy to see that, as long as the columns of Q form a basis of L , the sign of $Q^T A Q$ does not depend on the particular choice of the basis for L , so that the orientation of A on a subspace is well defined.

For arbitrarily chosen finite index sets I, J and vectors $a_i \in \mathbb{R}^n, b_j \in \mathbb{R}^n$ ($i \in I, j \in J$), let C be the polyhedral cone defined by

$$C := \{v \in \mathbb{R}^n \mid a_i^T v = 0 \ (i \in I), b_j^T v \leq 0 \ (j \in J)\}.$$

We recall that a *face* of C is any set which can be obtained as $H \cap C$, where H is a supporting hyperplane of C (a hyperplane H is a *supporting hyperplane* of C if C is entirely contained in one of the two closed half-spaces defined by H). We also recall that the intersection of faces is a face.

The following proposition collects some facts that will be useful in this paper.

PROPOSITION 2.1. *Let $\tilde{J}$ be a subset of J (the empty set is a possible subset). We define the following linear space:*

$$L(\tilde{J}) := \{v \in \mathbb{R}^n \mid a_i^T v = 0 \ (i \in I), b_j^T v = 0 \ (j \in \tilde{J})\}.$$

Then the following facts hold.

- (a) *For every $\tilde{J} \subseteq J$, $L(\tilde{J}) \cap C$ is a face of C .*
- (b) *If F is a face of C then there exists an index set $\tilde{J} \subseteq J$ such that $F = L(\tilde{J}) \cap C$.*
- (c) *Suppose that all the vectors a_i with $i \in I$ and b_j with $j \in J$ are linearly independent, and let $F = L(\tilde{J}) \cap C$ be a given face of C . Then the linear space generated by F is $L(\tilde{J})$.*

Proof. (a) $L(\tilde{J})$ is an intersection of supporting hyperplanes and hence $L(\tilde{J}) \cap C$, being the intersection of faces, is a face.

(b) This derives immediately from, e.g., Theorem 7.27 in [1]. For the proof of point (c) it is important to note that, according to Theorem 7.27 in [1], the set $\tilde{J}$ can be taken equal to $J(F)$, where $J(F) := \{j \in J \mid b_j^T v = 0 \ \forall v \in F\}$.

(c) Let $\tilde{J}$ be given, and let $F = L(\tilde{J}) \cap C$. By the observation we made in the proof of part (b), we know that $F = C \cap L(J(F))$. On the other hand, by Lemma 7.26

in [1], we also know that the linear space generated by F is $L(J(F))$. So to complete the proof it will suffice to show that $\tilde{J} = J(F)$. By definition we have

$$(2.1) \quad \begin{aligned} F &= C \cap L(\tilde{J}) \\ &= \{v \in \mathbb{R}^n \mid a_i^T v = 0 \ (i \in I), b_j^T v = 0 \ (j \in \tilde{J}), b_j^T v \leq 0 \ (j \in J \setminus \tilde{J})\}. \end{aligned}$$

On the other hand, we also have

$$(2.2) \quad \begin{aligned} F &= C \cap L(J(F)) \\ &= \{v \in \mathbb{R}^n \mid a_i^T v = 0 \ (i \in I), b_j^T v = 0 \ (j \in J(F)), b_j^T v \leq 0 \ (j \in J \setminus J(F))\}. \end{aligned}$$

Suppose now, by contradiction, that $\tilde{J} \neq J(F)$. By the definitions of F and $J(F)$, $\tilde{J} \subseteq J(F)$; this means that an index k exists which is in $J(F)$ but not in $\tilde{J}$. Since the vectors a_i with $i \in I$ and b_j with $j \in J$ are linearly independent, we can find a point $\bar{v}$ such that

$$a_i^T \bar{v} = 0 \ (i \in I), \quad b_j^T \bar{v} = 0 \ (j \in \tilde{J}), \quad b_j^T \bar{v} = -1 \ (j \in J \setminus \tilde{J}).$$

By (2.1) $\bar{v}$ belongs to F , but this contradicts (2.2) since we should have $b_k^T \bar{v} = 0$, and thus the proof is complete. $\square$

We shall now introduce the concept of a *coherently oriented* matrix over a cone and some related results; see [2, 21, 29, 30] and references therein.

DEFINITION 2.2. A matrix $A \in \mathbb{R}^{n \times n}$ is said to be *positively coherently oriented* (positively semicoherently oriented, negatively coherently oriented, negatively semicoherently oriented) on a polyhedral cone C if the orientation of the section of A in all linear spaces spanned by the faces of C is $+1$ ($+1$ or 0 , -1 , -1 or 0).

A is said to be *coherently oriented* (semicoherently oriented) on C if A is either positively coherently oriented (semicoherently oriented) or negatively coherently oriented (semicoherently oriented) on C .

It may be interesting to stress that the property of, let us say, positive coherent orientation is not hereditary, i.e., if A is positively coherently oriented on C and $C' \subseteq C$, then this does not imply that A is positively coherently oriented on C' . This is intuitively obvious since the property of being positively coherently oriented is connected to the matrix A and to the structure of the faces of C , and a cone C' contained in C can have a completely different structure. However, the following result, of which we omit the elementary proof, holds.

PROPOSITION 2.3. Suppose that A is positive definite (positive semidefinite) on a subspace L . If C is a cone contained in L , then A is positively coherently oriented (positively semicoherently oriented) on C .

We note that the property of positive coherent orientation differs and is more general than that of positive definiteness. However, it may be interesting to note that if the linear transformation A can be represented by a symmetric matrix, then the positive coherent orientation of A on a polyhedral convex cone containing no lines reduces to positive definiteness; see [31].

We finally restate a result due to Liu [21, Lemma 3.6], which will be used in our analysis of section 3 to determine the orientation of a matrix over a subspace.

LEMMA 2.4. Consider the square matrix

$$M = \begin{pmatrix} N & B \\ -B^T & 0 \end{pmatrix},$$

where $N \in \mathbb{R}^{n \times n}$ is arbitrary and $B \in \mathbb{R}^{n \times m}$, with $m \leq n$, has full column rank. Then, the orientation of M is equal to the orientation of the section of N in the subspace $\text{Ker}(B^T)$.

3. Nonsingularity conditions for $\partial\phi$. In this section, we present several conditions guaranteeing that all elements, or at least one element, in the generalized Jacobian $\partial\Phi(w)$ are nonsingular. The results of this section are of interest since, on the one hand, they relate the nonsingularity of $\partial\Phi$ to Robinson's [28] strong regularity condition and, on the other hand, they allow us to define superlinear convergent Newton methods for the solution of the nonlinear system $\Phi(w) = 0$; see [9, 11] for more details.

Our first aim is to better understand the structure of the generalized Jacobian of Φ . The next result gives some insight into the structure of the generalized Jacobian $\partial\Phi(w)$.

PROPOSITION 3.1. *Suppose that F is a C^1 -function and g and h are C^2 -functions. Let $w = (x, y, z) \in \mathbb{R}^{n+p+m}$. Then, each element $H \in \partial\Phi(w)^T$ can be represented as follows:*

$$H = \begin{pmatrix} \nabla_x L(w) & \nabla h(x) & \nabla g(x) D_a(w) \\ \nabla h(x)^T & 0 & 0 \\ -\nabla g(x)^T & 0 & D_b(w) \end{pmatrix},$$

where $D_a(w) := \text{diag}(a_1(w), \dots, a_m(w))$, $D_b(w) := \text{diag}(b_1(w), \dots, b_m(w)) \in \mathbb{R}^{m \times m}$ are diagonal matrices whose i th diagonal elements are given by

$$a_i(w) = \frac{g_i(x)}{\sqrt{g_i(x)^2 + z_i^2}} - 1, \quad b_i(w) = \frac{z_i}{\sqrt{g_i(x)^2 + z_i^2}} - 1$$

if $(g_i(x), z_i) \neq 0$ and by

$$a_i(w) = \xi_i - 1, \quad b_i(w) = \rho_i - 1 \quad \text{for any } (\xi_i, \rho_i) \text{ with } \|(\xi_i, \rho_i)\| \leq 1$$

if $(g_i(x), z_i) = 0$.

Proof. The first $n + p$ components of the vector function Φ are continuously differentiable, so the expression for the first $n + p$ columns of H readily follows. Then, consider the last m columns. Known rules on the evaluation of the generalized Jacobian (see [3, Proposition 2.6.2 (e)]) yield

$$\partial\phi(g(x), z)^T \subseteq \partial\phi(g_1(x), z_1) \times \dots \times \partial\phi(g_m(x), z_m).$$

If i is such that $(g_i(x), z_i) \neq (0, 0)$, then φ is continuously differentiable at $(g_i(x), z_i)$ and, again, the expression of the $(n + p + i)$ th column of H readily follows. If instead $(g_i(x), z_i) = (0, 0)$ then, using the theorem on the generalized gradient of a composite function (see [3, Theorem 2.3.9 (iii)]) and recalling that the generalized gradient of the Euclidean norm evaluated at the origin is the closed unit ball, we get

$$(3.1) \quad \partial\phi(g_i(x), z_i)^T \subseteq \{((\xi_i - 1)\nabla g_i(x)^T, 0, \dots, 0, (\rho_i - 1)e_i^T) \mid \|(\xi_i, \rho_i)\| \leq 1\}.$$

Hence, the proposition follows from the first part of the proof, (3.1), and the definition of $a_i(w)$ and $b_i(w)$. $\square$

The results of this section are strongly related to Robinson's definition of a strongly regular KKT point [28]. The original definition of strong regularity is somewhat involved; since we will not use it explicitly, we do not report it here. Instead, we shall take advantage of some equivalent definitions due to Robinson [28] and Liu [21]. In order to state these equivalent formulations, we need some more notation, in particular index set definitions. These index sets depend on the point at hand,

$w = (x, y, z) \in \mathbb{R}^{n+p+m}$. Some of them also depend on the particular element of the generalized Jacobian $\partial\Phi(w)$ at hand. This dependency will always be clear from the context and is therefore not explicitly indicated. Now let us define the index sets

$$I := \{1, \dots, m\},$$

$$I_0 := \{i \in I \mid g_i(x) = 0, z_i \geq 0\}, \quad I_> := \{i \in I \mid g_i(x) > 0, z_i = 0\}, \quad I_R := I \setminus (I_0 \cup I_>).$$

Moreover, we also need the further index sets

$$I_{00} := \{i \in I_0 \mid z_i = 0\}, \quad I_+ := \{i \in I_0 \mid z_i > 0\},$$

$$I_{01} := \{i \in I_{00} \mid \rho_i = 1\}, \quad I_{02} := \{i \in I_{00} \mid \rho_i < 1, \xi_i < 1\}, \quad I_{03} := \{i \in I_{00} \mid \xi_i = 1\},$$

$$I_{R2} := I_R \cup I_{02}$$

(see Proposition 3.1 for the definitions of ξ_i and ρ_i ; these numbers depend on the element of the generalized Jacobian of $\Phi(w)$). The following relationships between these index sets can easily be seen to hold:

$$I = I_0 \cup I_> \cup I_R, \quad I_0 = I_{00} \cup I_+, \quad I_{00} = I_{01} \cup I_{02} \cup I_{03}.$$

Taking into account the definition of these index sets and Proposition 3.1, every element $H \in \partial\Phi(w)^T$ has the following structure (the dependence on w is suppressed for simplicity):

$$(3.2) \quad H = \begin{pmatrix} \nabla_x L & \nabla h & -\nabla g_+ & -\nabla g_{01} & \nabla g_{R2}(D_a)_{R2} & 0 & 0 \\ \nabla h^T & 0 & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_+^T & 0 & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_{01}^T & 0 & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_{R2}^T & 0 & 0 & 0 & (D_b)_{R2} & 0 & 0 \\ -\nabla g_{03}^T & 0 & 0 & 0 & 0 & -I & 0 \\ -\nabla g_>^T & 0 & 0 & 0 & 0 & 0 & -I \end{pmatrix},$$

where $(D_a)_{R2}$ and $(D_b)_{R2}$ are negative definite diagonal matrices. Note that we abbreviated g_{I_+} , etc. to g_+ , etc. in (3.2). This notation will be used frequently throughout the paper.

It will also be useful to define some matrices which turn out to be closely related to the strong regularity condition:

$$M(J) := \begin{pmatrix} \nabla_x L & \nabla h & \nabla g_+ & \nabla g_J \\ -\nabla h^T & 0 & 0 & 0 \\ -\nabla g_+^T & 0 & 0 & 0 \\ -\nabla g_J^T & 0 & 0 & 0 \end{pmatrix},$$

where J is any subset of $I_{00} \cup I_R$. We are now in the position to summarize some known equivalent conditions for a KKT point of $\text{VIP}(X, F)$ to be strongly regular. The reader is referred to Robinson [28] and Liu [21] for the proofs. We point out that, as a simple consequence of Theorem 3.6, we will be able to give a new characterization of strong regularity.

THEOREM 3.2. *Let $w^* = (x^*, y^*, z^*) \in \mathbb{R}^{n+p+m}$ be a KKT point of $\text{VIP}(X, F)$. Then the following assertions are equivalent:*

- (a) w^* is a strongly regular KKT point.
- (b) For every $J \subseteq I_{00}$, the matrices $M(J)$ have the same nonzero orientation.
- (c) The gradients $\nabla h_i(x^*)$ ($i \in K$) and $\nabla g_i(x^*)$ ($i \in I_0$) are linearly independent, and the matrix $\nabla_x L(w^*)$ is coherently oriented on the cone

$$C(w^*) := \{v \in \mathbb{R}^n \mid \nabla h(x^*)^T v = 0, \nabla g_{I_+}(x^*)^T v = 0, \nabla g_{I_{00}}(x^*)^T v \leq 0\}.$$

- (d) $M(\emptyset)$ is nonsingular and either I_{00} is empty or $M(I_{00})/M(\emptyset)$ is a P -matrix.
- By point (b) of this theorem the following definition seems very natural.

DEFINITION 3.3. The extended strong regularity (ESR) condition is said to hold at a point w if the sign of the determinant of $M(J)$

- (i) is a nonzero constant for all index sets J such that $J \subseteq I_{00}$;
- (ii) cannot have opposite signs if J is such that $J \subseteq I_{00} \cup I_R$ (but can possibly be 0).

More precisely, Definition 3.3 (i) means that

$$\text{sign}(\det M(J_1)) \cdot \text{sign}(\det M(J_2)) > 0$$

for every $J_1, J_2 \subseteq I_{00}$, whereas part (ii) is equivalent to

$$\text{sign}(\det M(J_1)) \cdot \text{sign}(\det M(J_2)) \geq 0$$

for every $J_1, J_2 \subseteq I_{00} \cup I_R$.

This definition and its name are motivated by the fact that if $w = w^*$ is a KKT point, then $I_R = \emptyset$, and the ESR condition amounts to the requirement that, for all $J \subseteq I_{00}$, the sign of $M(J)$ is equal to the same nonzero constant (-1 or $+1$). By Theorem 3.2 (b) this is equivalent to the strong regularity of w^* . Thus condition ESR can be viewed as an *algebraic extension* of the concept of strong regularity to nonstationary points.

One of the main results we can prove is the following theorem.

THEOREM 3.4. Let $w = (x, y, z) \in \mathbb{R}^{n+p+m}$ be given, and suppose that the ESR condition holds at w . Then all elements in the generalized Jacobian $\partial\Phi(w)$ are nonsingular.

Proof. Consider an arbitrary but fixed element in $\partial\Phi(w)^T$. This element has the structure indicated in (3.2) and is obviously nonsingular if and only if the following matrix is nonsingular:

$$\begin{pmatrix} \nabla_x L & \nabla h & \nabla g_+ & \nabla g_{01} & \nabla g_{R2} & 0 & 0 \\ -\nabla h^T & 0 & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_+^T & 0 & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_{01}^T & 0 & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_{R2}^T & 0 & 0 & 0 & (D_b)_{R2}(D_a)_{R2}^{-1} & 0 & 0 \\ -\nabla g_{03}^T & 0 & 0 & 0 & 0 & I & 0 \\ -\nabla g_{>}^T & 0 & 0 & 0 & 0 & 0 & I \end{pmatrix}.$$

In turn this matrix is nonsingular if and only if the following matrix is nonsingular:

$$(3.3) \quad \begin{pmatrix} \nabla_x L & \nabla h & \nabla g_+ & \nabla g_{01} & \nabla g_{R2} \\ -\nabla h^T & 0 & 0 & 0 & 0 \\ -\nabla g_+^T & 0 & 0 & 0 & 0 \\ -\nabla g_{01}^T & 0 & 0 & 0 & 0 \\ -\nabla g_{R2}^T & 0 & 0 & 0 & (D_b)_{R2}(D_a)_{R2}^{-1} \end{pmatrix}.$$

The matrix (3.3) can be written as the sum of the matrix $M(J)$, with $J = I_{01} \cup I_{R2}$, and the diagonal matrix (the dimensions of the blocks are the same as in (3.3))

$$D := \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & (D_b)_{R2}(D_a)_{R2}^{-1} \end{pmatrix}.$$

We now recall that, given a square matrix $\bar{A}$ of dimension r and a diagonal matrix $\bar{D}$ (with the same dimension), one has (see [4, p. 60])

$$\det(\bar{D} + \bar{A}) = \sum_{\alpha} \det \bar{D}_{\alpha\alpha} \det \bar{A}_{\bar{\alpha}\bar{\alpha}},$$

where the summation ranges over all subsets α of $\{1, \dots, r\}$ (with complements $\bar{\alpha} = \{1, \dots, r\} \setminus \alpha$), and where it is assumed that the determinant of an “empty” matrix is 1. Exploiting this formula, the determinant of (3.3) can be written as

$$(3.4) \quad \det M(J) + \sum_{\alpha \subseteq I_{R2}} \det D_{\alpha\alpha} \det M(J)_{\bar{\alpha}\bar{\alpha}},$$

where the first term corresponds to $\alpha = \emptyset$. Moreover, we have taken into account that if α contains an element which does not belong to I_{R2} , then the determinant of $D_{\alpha\alpha}$ is 0. Since the nonzero diagonal elements of the matrix D are all positive, it follows that the determinants of $D_{\alpha\alpha}$ in (3.4) are all positive. Then to show that the determinant of (3.3) is nonzero and hence to conclude the proof, it will now be sufficient to show that the determinants of $M(J)$ and of all $M(J)_{\bar{\alpha}\bar{\alpha}}$ in (3.4) never have opposite signs, and that at least one of them is nonzero. These matrices can be written as $M(J \setminus \alpha)$, with $\alpha \subseteq I_{R2}$ (empty set included). If $\alpha = I_{R2}$, then $J \setminus I_{R2} = I_{01} \subseteq I_{00}$ while, in general, for every $\alpha \subseteq I_{R2}$ we have $J \setminus \alpha \subseteq I_{00} \cup I_R$. The theorem then follows directly from the definition of the ESR condition. $\square$

Remark. This result has interesting algorithmic consequences. In particular it implies that the semismooth Newton method [24, 25] applied to the system of equations (1.4) is locally quadratically convergent to a KKT point w^* of $\text{VIP}(X, F)$ if w^* is strongly regular (see [9, 11] for details). Even in the particular case in which the variational inequality $\text{VIP}(X, F)$ arises from a nonlinear program with inequality constraints, strong regularity compares favorably with the assumptions required by existing superlinear convergent methods with linear equation subproblems.

We see that the ESR condition is a sufficient condition for the nonsingularity of all the elements of the generalized Jacobian of Φ . If the point at hand is a KKT point, then it turns out that this is also a necessary condition. To prove this result, we first need the following technical proposition.

PROPOSITION 3.5. *Let $w = (x, y, z) \in \mathbb{R}^{n+p+m}$ be an arbitrary vector and assume that all elements in the generalized Jacobian $\partial\Phi(w)$ are nonsingular. Then, given any subset $J \subseteq I_{00}$, there exists an element in $\partial\Phi(w)$ such that $I_{01} = J$, $I_{02} = \emptyset$, and $I_{03} = I_{00} \setminus J$.*

Proof. In the first part of the proof we show that the nonsingularity of all elements in $\partial\Phi(w)$ implies the linear independence of the gradients $\nabla g_i(x)$ ($i \in I_{00}$). To this end, let $\{x^k\}$ be any sequence converging to x , and define $w^k = (x^k, y^k, z^k)$, where

$y^k = y$ for all k and

$$z_i^k = \begin{cases} z_i & \text{if } i \in I \setminus I_{00}, \\ 1/k & \text{if } i \in I_{00} \text{ and } g_i(x^k) = 0, \\ \sqrt{|g_i(x^k)|} & \text{if } i \in I_{00} \text{ and } g_i(x^k) \neq 0. \end{cases}$$

Then it is easy to see that the sequence $\{w^k\} = \{(x^k, y^k, z^k)\} \subset \mathbb{R}^{n+p+m}$ converges to w such that Φ is differentiable at each w^k , and that $\{\nabla\Phi(w^k)\}$ converges to an element $H \in \partial\Phi(w)^T$ such that $I_{01} = I_{00}$ and $I_{02} = I_{03} = \emptyset$. But then, since H is nonsingular by assumption, the linear independence of the gradients $\nabla g_i(x)$ ($i \in I_{00}$) follows immediately from the general structure of the (transposed) generalized Jacobian given in (3.2).

For the second part of the proof, let $J \subseteq I_{00}$ be any given subset. We now build another sequence $\{w^k\} = \{(x^k, y^k, z^k)\} \subset \mathbb{R}^{n+p+m}$ converging to w such that Φ is differentiable at each w^k and $\{\nabla\Phi(w^k)\}$ converges to an element $H \in \partial\Phi(w)^T$ such that H enjoys the properties stated in the proposition. By the first part of the proof, we know that the gradients $\nabla g_i(x)$ ($i \in I_{00}$) are linearly independent. Hence it follows from the implicit function theorem that there is a sequence $\{x^k\}$ converging to x such that $g_i(x^k) > 0$ for all $i \in I_{00}$ and all k . Then take $w^k = (x^k, y^k, z^k)$, with x^k being the k th iterate of the above sequence, $y^k = y$ for all k , and

$$z_i^k = \begin{cases} z_i & \text{if } i \in I \setminus I_{00}, \\ \sqrt{g_i(x^k)} & \text{if } i \in J, \\ g_i(x^k)^2 & \text{if } i \in I_{00} \setminus J. \end{cases}$$

Note that $\{w^k\}$ tends to w and that Φ is differentiable at w^k , because there is no index i such that the case $g_i(x^k) = z_i^k = 0$ occurs. Furthermore, using Proposition 3.1 and the continuity of the functions involved, it easily follows that $\{\nabla\Phi(w^k)\}$ converges to an element H having the desired properties. $\square$

THEOREM 3.6. *Let $w^* = (x^*, y^*, z^*) \in \mathbb{R}^{n+p+m}$ be a KKT point of $VIP(X, F)$. Then the ESR condition is satisfied at w^* if and only if all elements in the generalized Jacobian $\partial\Phi(w^*)$ are nonsingular.*

Proof. The necessity part of the theorem is just Theorem 3.4. We therefore turn to the sufficiency part.

Assume that all elements in the generalized Jacobian $\partial\Phi(w^*)$ are nonsingular. We first prove that the nonzero orientation of all these matrices is the same. In fact, suppose, by contradiction, that there are $H_1, H_2 \in \partial\Phi(w^*)^T$ such that $\det(H_1) < 0$ and $\det(H_2) > 0$. Since the determinant is a continuous function, a convex combination $\bar{H}$ of H_1 and H_2 exists such that $\det(\bar{H}) = 0$. Because $\partial\Phi(w^*)$ is convex by definition, the singular matrix $\bar{H}$ belongs to $\partial\Phi(w^*)^T$, thus contradicting the nonsingularity of all elements in the generalized Jacobian $\partial\Phi(w^*)$.

Because w^* is a KKT point, the subset I_R must be empty. Therefore, to be sure that the ESR condition holds, we need only that the matrices $M(J)$ have the same nonzero orientation for all $J \subseteq I_{00}$. According to Proposition 3.5, for each $J \subseteq I_{00}$, we can find an $H(J) \in \partial\Phi(w^*)^T$ such that $J = I_{01}$ and $I_{03} = I_{00} \setminus I_{01}$. With regard to the structure of the elements of $\partial\Phi(w^*)^T$ indicated in (3.2), and recalling that

$h : \mathbb{R}^n \rightarrow \mathbb{R}^p$ and $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$, we obtain

$$0 \neq \text{sign}(\det H(J)) = (-1)^{p+m} \text{sign} \left(\det \begin{pmatrix} \nabla_x L & \nabla h & \nabla g_+ & \nabla g_J & 0 & 0 \\ -\nabla h^T & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_+^T & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_J^T & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_{03}^T & 0 & 0 & 0 & I & 0 \\ -\nabla g_{>}^T & 0 & 0 & 0 & 0 & I \end{pmatrix} \right),$$

which, in turn, implies

$$\begin{aligned} 0 \neq \text{sign}(\det H(J)) &= (-1)^{p+m} \text{sign} \left(\det \begin{pmatrix} \nabla_x L & \nabla h & \nabla g_+ & \nabla g_J \\ -\nabla h^T & 0 & 0 & 0 \\ -\nabla g_+^T & 0 & 0 & 0 \\ -\nabla g_J^T & 0 & 0 & 0 \end{pmatrix} \right) \\ &= (-1)^{p+m} \text{sign}(\det M(J)). \end{aligned}$$

Since, in view of the first part of this proof, all matrices $H \in \partial\Phi(w^*)^T$ have the same nonzero orientation, it follows that the ESR condition is satisfied. $\square$

As we already observed, if $w = w^*$ is a KKT point, then the ESR condition is equivalent to the strong regularity condition. We therefore get as an immediate consequence of Theorem 3.6 the following new characterization of a strongly regular KKT point.

COROLLARY 3.7. *A KKT point w^* of $\text{VIP}(X, F)$ is strongly regular if and only if all the elements in the generalized Jacobian $\partial\Phi(w^*)$ are nonsingular.*

We now illustrate with some examples the results obtained so far.

Example 1. Consider $\text{VIP}(X, F)$ with $n = 1$, $p = 0$, $m = 1$, $F(x) = x - 1$, and $g_1(x) = x$. For this problem we have

$$\Phi(x, z) = \left(\frac{x - 1 - z}{\sqrt{x^2 + z^2} - x - z} \right).$$

Consider the point $(x, z) = (0, 0)$; in this point $I_{00} = \{1\}$ while $I_+ = I_R = \emptyset$. We have

$$\det M(\emptyset) = \det(1) = 1, \quad \det M(\{1\}) = \det \begin{pmatrix} 1 & 1 \\ -1 & 0 \end{pmatrix} = 1,$$

so that the ESR condition is satisfied. From the expression of Φ it can be easily seen that the elements of $\partial\Phi(0, 0)$ are given by

$$(3.5) \quad \begin{pmatrix} 1 & -1 \\ \alpha - 1 & \beta - 1 \end{pmatrix},$$

with $(\alpha, \beta)^T$ belonging to the generalized gradient of $\sqrt{x^2 + z^2}$ in $(0, 0)$. Therefore, $\|(\alpha, \beta)\| \leq 1$ so that all the matrices in (3.5) are nonsingular, as expected.

Example 2. Consider $\text{VIP}(X, F)$ with $n = 1$, $p = 0$, $m = 1$, $F(x) = -x$, and $g_1(x) = x$. For this problem we have

$$\Phi(x, z) = \left(\frac{-x - z}{\sqrt{x^2 + z^2} - x - z} \right).$$

Consider the KKT point $(x, z) = (0, 0)$; in this point $I_{00} = \{1\}$ while $I_+ = I_R = \emptyset$. We have

$$\det(M(\emptyset)) = \det(-1) = -1, \quad \det(M(\{1\})) = \det \begin{pmatrix} -1 & 1 \\ -1 & 0 \end{pmatrix} = 1,$$

so that the ESR condition is not satisfied. From the expression of Φ it can be easily seen that the elements of $\partial\Phi(0, 0)$ are given by

$$(3.6) \quad \begin{pmatrix} -1 & -1 \\ \alpha - 1 & \beta - 1 \end{pmatrix},$$

with $(\alpha, \beta)^T$ belonging to the generalized gradient of $\sqrt{x^2 + z^2}$ in $(0, 0)$. If we take $\alpha = \beta$ such that $\|(\alpha, \beta)\| \leq 1$, e.g., $\alpha = \beta = 1/\sqrt{2}$, we see that the corresponding element in (3.6) is singular, whereas it is nonsingular whenever $\alpha \neq \beta$.

Motivated by Theorem 3.2, our aim is now to obtain conditions which are necessary and sufficient for ESR to hold by extending those of points (c) and (d) in Theorem 3.2.

We shall first consider conditions extending (c). To this end, we introduce the notation

$$C(w; I_1) := \{v \in \mathbb{R}^n \mid \nabla h(x)^T v = 0, \nabla g_{I_+}(x)^T v = 0, \nabla g_{I_1}(x)^T v \leq 0\},$$

where I_1 is a subset of $I \setminus I_+$. Note that, with this notation, the polyhedral cone in point (c) of Theorem 3.2 corresponds to $C(w; I_{00})$.

THEOREM 3.8. *Let $w = (x, y, z) \in \mathbb{R}^{n+p+m}$ be given. Then the ESR condition is satisfied at w if and only if the following three conditions hold:*

- (a) *The vectors $\nabla h_i(x)$ ($i \in K$) and $\nabla g_i(x)$ ($i \in I_0$) are linearly independent.*
- (b) *The matrix $\nabla_x L(w)$ is positively (negatively) coherently oriented over $C(w; I_{00})$.*
- (c) *The matrix $\nabla_x L(w)$ is positively (negatively) semicoherently oriented over $C(w; I_{00} \cup \tilde{I})$ for every $\tilde{I} \subseteq I_R$ such that the vectors $\nabla h_i(x)$ ($i \in K$) and $\nabla g_i(x)$ ($i \in I_0 \cup \tilde{I}$) are linearly independent.*

Proof. By Proposition 2.1 (a), (b), all and only the faces of $C(w; I_{00})$ can be obtained as $L(J) \cap C(w; I_{00})$, for all $J \subseteq I_{00}$, where

$$L(J) := \{v \in \mathbb{R}^n \mid \nabla h(x)^T v = 0, \nabla g_{I_+}(x)^T v = 0, \nabla g_J(x)^T v = 0\}.$$

Then, by condition (a) and Proposition 2.1 (c), all and only the linear subspaces generated by the faces of $C(w; I_{00})$ can be obtained as $L(J)$, for all $J \subseteq I_{00}$. Taking into account that $L(J) = \text{Ker}((\nabla h(x), \nabla g_{I_+}(x), \nabla g_J(x))^T)$ for $J \subseteq I_{00}$, we have, by Lemma 2.4 (take $B = (\nabla h(x), \nabla g_{I_+}(x), \nabla g_J(x))$) and (a), that the sign of the determinant of $M(J)$ is equal to that of the section of $\nabla_x L(w)$ in the subspace $L(J)$, so that part (i) of the definition of ESR follows by (b).

Assume now that $J \subseteq I_{00} \cup I_R$. If the vectors $\nabla h_i(x)$ ($i \in K$) and $\nabla g_i(x)$ ($i \in I_+ \cup J$) are not linearly independent, then $M(J)$ is obviously singular. If, otherwise, the set of vectors is linearly independent, then we can consider the cone $C(w; J)$ and reason as before so that, by (c), we have that (ii) of ESR holds.

The opposite implication can easily be obtained by reversing the arguments given above. $\square$

Remark. In the statement of Theorem 3.8 we have stressed the fact that the orientations on the faces of the cones considered in points (b) and (c) have to be the same when not 0; however, this is superfluous, since this fact holds automatically.

In condition (c) of Theorem 3.8 we could have equivalently considered only those subsets $\tilde{I}$ for which there is no other subset $\tilde{I}'$ such that $\tilde{I} \subseteq \tilde{I}' \subseteq I_R$ and the vectors $\nabla h_i(x)$ ($i \in K$) and $\nabla g_i(x)$ ($i \in I_0 \cup \tilde{I}'$) are linearly independent. In particular, if the vectors $\nabla h_i(x)$ ($i \in K$) and $\nabla g_i(x)$ ($i \in I_0 \cup I_R$) are linearly independent, then we could just consider the cone $C(w; I_{00} \cup I_R)$.

As a corollary, we get the following useful result.

COROLLARY 3.9. *Let $w = (x, y, z) \in \mathbb{R}^{n+p+m}$ be any given vector. Assume that*

- (a) *the gradients $\nabla h_i(x)$ ($i \in K$) and $\nabla g_i(x)$ ($i \in I_0$) are linearly independent;*
- (b) *$v^T \nabla_x L(w) v > 0$ for all $v \in \mathbb{R}^n, v \neq 0$, such that $\nabla h_i(x)^T v = 0$ ($i \in K$) and $\nabla g_i(x)^T v = 0$ ($i \in I_+$).*

Then all elements $H \in \partial\Phi(w)$ are nonsingular.

Proof. Condition (a) of Corollary 3.9 is obviously the same as (a) of Theorem 3.8. On the other hand, condition (b) of the corollary requires that $\nabla_x L(w)$ be positive definite on a subspace that contains all the cones considered in assumptions (b) and (c) of Theorem 3.8. By Proposition 2.3 this implies that (b) and (c) of Theorem 3.8 are fulfilled. $\square$

Note that, if $w = w^*$ is a KKT point of $\text{VIP}(X, F)$, then the above corollary reduces to a recent result by Jiang [18]; see also Fischer [14].

We now consider necessary and sufficient conditions for ESR to hold that are inspired by (d) in Theorem 3.2.

THEOREM 3.10. *The ESR condition is satisfied in a point $w = (x, y, z) \in \mathbb{R}^{n+p+m}$ if and only if the following three conditions hold:*

- (a) *$M(\emptyset)$ is nonsingular.*
- (b) *Either I_{00} is empty or $M(I_{00})/M(\emptyset)$ is a P -matrix.*
- (c) *Either $I_{00} \cup I_R$ is empty or $M(I_{00} \cup I_R)/M(\emptyset)$ is a P_0 -matrix.*

Proof. We only consider the case that $M(\emptyset)$ is nonsingular since, otherwise, neither the ESR condition nor (a) of Theorem 3.10 is satisfied. The principal submatrices of $M(I_{00})/M(\emptyset)$ are exactly the matrices given by $M(J)/M(\emptyset)$, for all $J \subseteq I_{00}$. Using the determinantal formula for the Schur-complement, we get

$$\det M(J) = \det (M(J)/M(\emptyset)) \det M(\emptyset).$$

Hence the sign of $\det M(J)$ is equal to the sign of $\det M(\emptyset)$ for all $J \subseteq I_{00}$ if and only if all the principal minors of $M(I_{00})/M(\emptyset)$ are positive. This fact establishes the equivalence of condition (b) in Theorem 3.10 and of (i) in the ESR condition. The equivalence of (c) with (ii) can be shown in a similar way. We omit the details in this paper. $\square$

In some cases (see, e.g., Proposition 4.2) it may also be of interest to determine conditions ensuring that at least one element in the generalized Jacobian $\partial\Phi(w)$ is nonsingular. We devote the last part of this section to this issue. We naturally expect that a weaker condition than ESR will be needed for establishing this result. Actually, the following condition will be useful.

DEFINITION 3.11. *The extended weak regularity (EWR) condition is said to hold at a point $w = (x, y, z) \in \mathbb{R}^{n+p+m}$ if a possibly empty index set $\tilde{J}$, with $\tilde{J} \subseteq I_{00}$, exists such that*

- (i) *the sign of the determinant of $M(\tilde{J})$ is $+1$ (-1);*
- (ii) *each sign of the determinants of $M(J)$ is either 0 or $+1$ (-1) for all $J \subseteq I_{00} \cup I_R$.*

Note that the EWR and the ESR conditions coincide if $I_{00} = \emptyset$. In particular, if $w = w^*$ is a KKT point, the two conditions coincide if strict complementarity

holds. In general, however, the EWR condition is weaker than the ESR condition. We illustrate this by a simple example.

Example 3. Consider $\text{VIP}(X, F)$ with $n = 2$, $p = 0$, $m = 1$, $F(x) := (x_1, x_2)^T$, and $g_1(x) = x_2$. Consider the KKT point $(x_1, x_2, z) = (0, 0, 0)$. We have $I_{00} = \{1\}$, while $I_+ = I_R = \emptyset$. We have

$$M(\emptyset) = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad M(\{1\}) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix}.$$

Therefore, $\det(M(\emptyset)) = 0$ and $\det(M(\{1\})) = 1$, so that the ESR condition is not satisfied, whereas the EWR condition obviously holds by taking $\tilde{J} = \{1\}$.

The following result is the direct counterpart of Theorem 3.8 and gives a complete characterization of the EWR condition in terms of the (semi)coherent orientation of a certain matrix.

THEOREM 3.12. *Let $w = (x, y, z) \in \mathbb{R}^{n+p+m}$ be given. Then the EWR condition is satisfied at w if and only if there is a (possibly empty) index set $\tilde{J} \subseteq I_{00}$ such that the following three conditions hold:*

- (a) *The gradients $\nabla h_i(x)$ ($i \in K$) and $\nabla g_i(x)$ ($i \in I_+ \cup \tilde{J}$) are linearly independent.*
- (b) *The orientation of the matrix $\nabla_x L(w)$ on the subspace*

$$\{v \in \mathbb{R}^n \mid \nabla h(x)^T v = 0, \nabla g_{I_+}(x)^T v = 0, \nabla g_{\tilde{J}}(x)^T v = 0\}$$

is positive (negative).

- (c) *The matrix $\nabla_x L(w)$ is positively (negatively) coherently oriented over $C(w; I_{00} \cup \tilde{I})$ for every $\tilde{I} \subseteq I_R$ such that the vectors $\nabla h_i(x)$ ($i \in K$) and $\nabla g_i(x)$ ($i \in I_0 \cup \tilde{I}$) are linearly independent.*

Proof. The proof is almost identical to the one of Theorem 3.8, so we omit the details. $\square$

Moreover, we can prove the following result.

THEOREM 3.13. *Let $w = (x, y, z) \in \mathbb{R}^{n+p+m}$ be given, and suppose that the EWR condition holds at w . Then the generalized Jacobian $\partial\Phi(w)$ contains at least one nonsingular element.*

Proof. Let $\tilde{J} \subseteq I_{00}$ denote any index set which satisfies conditions (i) and (ii) in Definition 3.11. We first show that there is a matrix $\tilde{H} \in \partial\Phi(w)^T$ with $I_{01} = \tilde{J}$. To this end, let us define a sequence $\{w^k\} = \{(x^k, y^k, z^k)\} \subset \mathbb{R}^{n+p+m}$ such that $(x^k, y^k) := (x, y)$ for all k and such that the sequence $\{z^k\}$ satisfies the conditions $z^k \rightarrow z$ and, for all k , $z_i^k > 0$ if $i \in \tilde{J}$, $z_i^k < 0$ if $i \in I_{00} \setminus \tilde{J}$, and $z_i^k = z_i$ if $i \in I \setminus I_{00}$. Note that, since $\tilde{J} \subseteq I_{00}$, it is always possible to find a sequence $\{z^k\}$ satisfying these conditions. Then it can easily be verified that $\{w^k\}$ converges to w , that Φ is differentiable at each w^k , and that we can assume, renumbering if necessary, that the sequence $\{\nabla\Phi(w^k)\}$ converges to a certain element $\tilde{H} \in \partial\Phi(w)^T$. Taking into account Proposition 3.1 it further follows that, for each w^k , $b_i(w^k) = 0$ if and only if $i \in \tilde{J}$. With regard to the definition of I_{01} we obtain that $I_{01} = \tilde{J}$ is valid for $\tilde{H}$.

The proof of the theorem is now almost a repetition of that of Theorem 3.4 in which we take into account that $I_{01} = \tilde{J}$. We report here the complete proof for the sake of clarity.

We shall show that the particular matrix $\tilde{H}$ constructed above is nonsingular. This matrix has the structure indicated in (3.2), and obviously it is nonsingular if and

only if the following matrix is nonsingular:

$$\begin{pmatrix} \nabla_x L & \nabla h & \nabla g_+ & \nabla g_{\tilde{J}} & \nabla g_{R2} & 0 & 0 \\ -\nabla h^T & 0 & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_+^T & 0 & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_{\tilde{J}}^T & 0 & 0 & 0 & 0 & 0 & 0 \\ -\nabla g_{R2}^T & 0 & 0 & 0 & (D_b)_{R2}(D_a)_{R2}^{-1} & 0 & 0 \\ -\nabla g_{03}^T & 0 & 0 & 0 & 0 & I & 0 \\ -\nabla g_{>}^T & 0 & 0 & 0 & 0 & 0 & I \end{pmatrix},$$

where, we recall, $I_{01} = \tilde{J}$ holds. Reasoning as in the proof of Theorem 3.4, we can see that this matrix is nonsingular if and only if

$$(3.7) \quad \det M(J) + \sum_{\alpha \subseteq I_{R2}} \det D_{\alpha\alpha} \det M(J)_{\bar{\alpha}\bar{\alpha}} \neq 0,$$

with $J = \tilde{J} \cup I_{R2}$. Again, as in the proof of Theorem 3.4, to show that (3.7) holds it will now be sufficient to show that the determinants of $M(J)$ and of all $M(J)_{\bar{\alpha}\bar{\alpha}}$ in (3.7) never assume opposite signs, and that at least one of them is nonzero. These matrices can be written as $M(J \setminus \alpha)$, with $\alpha \subseteq I_{R2}$ (empty set included). If $\alpha = I_{R2}$, then $J \setminus I_{R2} = I_{01} = \tilde{J}$, while, in general, for every $\alpha \subseteq I_{R2}$ we have $J \setminus \alpha \subseteq I_{00} \cup I_R$. The thesis then follows directly from the EWR condition (Definition 3.11). $\square$

Example 4. Consider the problem introduced in Example 3. We have

$$\Phi(x_1, x_2, z) = \begin{pmatrix} x_1 \\ x_2^2 - z \\ \sqrt{x_2^2 + z^2} - x_2 - z \end{pmatrix},$$

so that, in $(x_1, x_2, z) = (0, 0, 0)$, we have

$$\partial\Phi(0, 0, 0) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & \alpha - 1 & \beta - 1 \end{pmatrix},$$

where $(\alpha, \beta)^T$ belongs to the generalized gradient of $\sqrt{x_2^2 + z^2}$ in $(0, 0)$. It is then easy to see that $\partial\Phi(0, 0, 0)$ contains a nonsingular element; for example, take $(\alpha, \beta) = (0, 1)$. On the other hand, it also contains a singular element; to this end, take $(\alpha, \beta) = (1, 0)$.

Remark. We proved in Theorem 3.6 that the ESR condition is satisfied in a KKT point w if and only if all the elements in $\partial\Phi(w)$ are nonsingular. On the other hand, we proved in Theorem 3.13 that, if the EWR condition is satisfied in a KKT point w , at least one nonsingular element exists in $\partial\Phi(w)$. It is then natural to ask whether the existence of a nonsingular element in $\partial\Phi(w)$ implies the EWR condition. The answer is no, and this is shown by Example 2. In fact, the problem in this example does not satisfy the EWR condition in the KKT point $(x, z) = (0, 0)$ considered there. However, it is sufficient to consider $(\alpha, \beta) = (0, 1)$ in (3.6) to obtain a nonsingular generalized Jacobian of $\partial\Phi(0, 0)$.

We next state a simple consequence of Theorem 3.13.

COROLLARY 3.14. *Let $w = (x, y, z) \in \mathbb{R}^{n+p+m}$ be an arbitrary vector. Assume that*

- (a) *the gradients $\nabla h_i(x)$ ($i \in K$) and $\nabla g_i(x)$ ($i \in I_+$) are linearly independent;*

- (b) $v^T \nabla_x L(w)v > 0$ for all $v \in \mathbb{R}^n, v \neq 0$, such that $\nabla h_i(x)^T v = 0$ ($i \in K$) and $\nabla g_i(x)^T v = 0$ ($i \in I_+$).

Then there is at least one nonsingular element in $\partial\Phi(w)$.

Proof. We first show that assumptions (a) and (b) imply that the EWR condition is satisfied at w . For $\tilde{J} := \emptyset$ assumption (a) of Theorem 3.12 is equivalent to assumption (a) of Corollary 3.14. Furthermore, all the cones which appear in assumptions (b) and (c) of Theorem 3.12 are contained in the linear space

$$\{v \in \mathbb{R}^n \mid \nabla h(x)^T v = 0, \nabla g_{I_+}(x)^T v = 0\}.$$

Hence, by Proposition 2.3, assumptions (b) and (c) of Theorem 3.12 follow from assumption (b) of this Corollary. Therefore, the EWR condition is satisfied at w because of Theorem 3.12. Thus the assertion follows immediately from Theorem 3.13. $\square$

Some explanations for Corollaries 3.9 and 3.14 are in order. If the arbitrary vector w in these results is a KKT point of $\text{VIP}(X, F)$ and if we borrow some terminology from the optimization literature, then Corollary 3.9 states that all elements in the generalized Jacobian $\partial\Phi(w)$ are nonsingular if

- (a) the gradients of all active constraints are linearly independent, and
- (b) the strong second-order condition is satisfied.

On the other hand, Corollary 3.14 guarantees the existence of at least one nonsingular element in $\partial\Phi(w)$ if condition (a) is replaced by the weaker assumption that only the gradients belonging to the strongly active constraints are linearly independent.

We stress, however, that the vector w in our results is an arbitrary vector, not necessarily a KKT point, and that we have obtained these corollaries as simple consequences of more general results.

In the following proposition, we state that if the strong second-order condition is weakened to the second-order condition, but if the gradients of all binding constraints are linearly independent, then the assertion of Corollary 3.14 remains true. The proof of this result is omitted for the sake of brevity.

PROPOSITION 3.15. *Let $w = (x, y, z) \in \mathbb{R}^{n+p+m}$ be any given vector. Assume that*

- (a) *the gradients $\nabla h_i(x)$ ($i \in K$) and $\nabla g_i(x)$ ($i \in I_0$) are linearly independent;*
- (b) *$v^T \nabla_x L(w)v > 0$ for all $v \in \mathbb{R}^n, v \neq 0$, such that $\nabla h_i(x)^T v = 0$ ($i \in K$) and $\nabla g_i(x)^T v = 0$ ($i \in I_0$).*

Then there is at least one nonsingular element $H \in \partial\Phi(w)$.

4. Stationary conditions for ψ . In this section, after noting that Ψ is continuously differentiable, we study conditions which guarantee that $\nabla\Psi(w^*) = 0$ implies $\Psi(w^*) = 0$, i.e., that a stationary point of the merit function Ψ is a KKT point of $\text{VIP}(X, F)$. After considering the general case we will deal with variational inequalities on polyhedral sets X for which we are able to obtain stronger results. These conditions play a key role in the algorithms which seek KKT points of $\text{VIP}(X, F)$ through the minimization of Ψ . In fact, by using unconstrained minimization algorithms we can only hope to find stationary points of Ψ . Hence, the weaker the conditions which guarantee that a stationary point of Ψ is a global solution, the more effective the unconstrained minimization of Ψ will be.

As a first step, we state the smoothness of Ψ and give an expression of its gradient.

PROPOSITION 4.1. *If F is a C^1 -function and g, h are C^2 -functions, then Ψ is continuously differentiable and $\nabla\Psi(w) = H\Phi(w)$ for every $H \in \partial\Phi(w)^T$.*

Proof. By the chain rule for the composition of a strictly differentiable function and a Lipschitz-continuous function (see [3, Theorem 2.6.6]) we have $\partial\Psi(w) = \partial\Phi(w)^T\Phi(w)$. It is easy to check that $\partial\Phi(w)^T\Phi(w)$ is single valued everywhere because the zero components of $\Phi(w)$ cancel the “multivalued columns” of $\partial\Phi(w)^T$. Therefore we have, using the corollary to Theorem 2.2.4 in [3], that Ψ is continuously differentiable and that $\nabla\Psi(w) = H\Phi(w)$ for every $H \in \partial\Phi(w)^T$. $\square$

4.1. General X. We first state a simple but useful consequence of Proposition 4.1.

PROPOSITION 4.2. *Let $w^* = (x^*, y^*, z^*) \in \mathbb{R}^{n+p+m}$ be a stationary point of Ψ . Then w^* is a KKT point of $\text{VIP}(X, F)$ if there is at least one nonsingular element in the generalized Jacobian $\partial\Phi(w^*)$.*

Proof. Since w^* is a stationary point of Ψ , we have by Proposition 4.1 that

$$(4.1) \quad 0 = \nabla\Psi(w^*) = H\Phi(w^*)$$

for any $H \in \partial\Phi(w^*)^T$. But, by the assumptions made, we have that the matrix H can be chosen to be nonsingular. By (4.1), this implies $\Phi(w^*) = 0$, so that w^* is a KKT point of $\text{VIP}(X, F)$. $\square$

By the results of the previous section, this proposition implies, in particular, that if w^* is a stationary point of Ψ such that the ESR or the EWR condition holds at w^* , then w^* is a KKT point of the VIP. It is, however, possible to give further conditions which still guarantee that every stationary point of the function Ψ is a KKT point of $\text{VIP}(X, F)$.

THEOREM 4.3. *Let $w^* = (x^*, y^*, z^*) \in \mathbb{R}^{n+p+m}$ be a stationary point of Ψ . Assume that*

- (a) $\nabla_x L(w^*)$ *is positive semidefinite on $\mathbb{R}^n$;*
- (b) $\nabla_x L(w^*)$ *is positive definite on $\text{Ker}(\nabla h(w^*)^T) \cap \text{Ker}(\nabla g(w^*)^T)$;*

and either of the following two conditions holds:

- (c1) $\nabla h(x^*)$ *has full column rank;*
- (c2) h *is an affine function and X is nonempty.*

Then $\Psi(w^*) = 0$; i.e., w^* is a KKT point of $\text{VIP}(X, F)$.

Proof. Suppose that $\nabla\Psi(w^*) = 0$. Using Propositions 4.1 and 3.1, this can be written as

$$(4.2) \quad \nabla_x L(w^*)L^* + \nabla h(x^*)h^* + \nabla g(x^*)D_a(w^*)\phi^* = 0,$$

$$(4.3) \quad \nabla h(x^*)^T L^* = 0,$$

$$(4.4) \quad -\nabla g(x^*)^T L^* + D_b(w^*)\phi^* = 0,$$

where (L^*, h^*, ϕ^*) denotes $(L(w^*), h(x^*), \phi(g(x^*), z^*))$. Moreover, if $\phi_i^* = 0$ then the values of the i th diagonal elements of the matrices $D_a(w^*)$ and $D_b(w^*)$ are immaterial to the value of the left-hand side of (4.2) and (4.4). Therefore we can assume, without loss of generality, that these diagonal elements are all equal to -1 . With this convention we have that the diagonal matrices $D_a(w^*)$ and $D_b(w^*)$ are nonsingular with negative diagonal elements (note that a diagonal element of these two matrices can be 0 only if the corresponding element of ϕ^* is equal to 0; compare Proposition 3.1).

We can now solve (4.4) with respect to ϕ^* and substitute in (4.2), thus obtaining

$$(4.5) \quad (\nabla_x L(w^*) + \nabla g(x^*)D\nabla g(x^*)^T)L^* + \nabla h(x^*)h^* = 0,$$

where we have indicated by D the positive diagonal matrix $D_a(w^*)D_b(w^*)^{-1}$. Premultiplying (4.5) by $(L^*)^T$ and taking (4.3) into account, we obtain

$$(4.6) \quad (L^*)^T(\nabla_x L(w^*) + \nabla g(x^*)D\nabla g(x^*)^T)L^* = 0.$$

Since D is positive definite and the matrix $\nabla g(x^*)D\nabla g(x^*)^T$ is positive semidefinite, it is easy to see, using assumptions (a) and (b) and the fact that $L^* \in \text{Ker}(\nabla h(x^*)^T)$ by (4.3), that (4.6) implies $L^* = 0$. Hence $\phi^* = 0$ follows from (4.4), and (4.5) becomes

$$(4.7) \quad \nabla h(x^*)h^* = 0.$$

We now want to show that $h^* = 0$. Assume first that (c1) holds. Then, since $\nabla h(x^*)$ has full column rank, we immediately get $h^* = h(x^*) = 0$ from (4.7). Assume now, instead, that (c2) holds. Since h is an affine function, the norm function $\frac{1}{2}\|h\|^2$ is convex with gradient equal to the left-hand side of (4.7). Hence (4.7) implies that x^* is a minimum point of the norm function $\frac{1}{2}\|h\|^2$. Since the set X is nonempty, this in turn implies $h^* = h(x^*) = 0$.

Therefore, in both cases, $(L^*, h^*, \phi^*) = 0$; hence $\Phi(w^*) = 0$, so that w^* is a KKT point of $\text{VIP}(X, F)$. $\square$

4.2. Polyhedral X . A particularly interesting case is when X is *polyhedral*, i.e.,

$$h(x) = Ax - b, \quad g(x) = Cx - d,$$

with $A \in \mathbb{R}^{p \times n}$, $b \in \mathbb{R}^p$, $C \in \mathbb{R}^{m \times n}$, and $d \in \mathbb{R}^m$. A first important consequence of the polyhedrality of X is that the KKT conditions (1.3) are both necessary and sufficient, so that any solution of the VIP corresponds to a certain solution of the nonsmooth system of equations (1.4) and vice versa. Furthermore, in this case some nice geometrical interpretations can be given to the conditions employed in Theorem 4.3. In particular we note that, in this case, $\text{Ker}(A) \cap \text{Ker}(C)$ is the lineality space of the set X . So, if this lineality space reduces to $\{0\}$, then conditions (a) and (b) of Theorem 4.3 collapse to the requirement that $\nabla_x L(w)$ is just positive semidefinite. This discussion can be summarized in the following theorem, where we have taken into account that in this case the Jacobian of the Lagrangian L is just $\nabla F(x)^T$.

THEOREM 4.4. *Assume that X is polyhedral. Let $w^* = (x^*, y^*, z^*) \in \mathbb{R}^{n+p+m}$ be a stationary point of Ψ . Then w^* is a solution of $\text{VIP}(X, F)$ if*

- (a1) $\nabla F(x^*)$ is positive semidefinite on $\mathbb{R}^n$;
- (b1) $\nabla F(x^*)$ is positive definite on the lineality space of X .

In particular, if either

- (1) *the set X is bounded, or*
- (2) *the set X is contained in the nonnegative orthant $\mathbb{R}_+^n$*

is satisfied, then assumption (a1) is sufficient for the theorem to hold.

Proof. (a1) and (b2) are just (a) and (b) of Theorem 4.3 rewritten, taking into account that, since the constraints are affine, then $\nabla_x L(w) = \nabla F(x)$, and recalling that the lineality space of X is $\text{Ker}(A) \cap \text{Ker}(C)$. The second part of the theorem follows by observing that the condition (1) or (2) obviously implies that the set X contains no lines and hence that the lineality space of X is $\{0\}$. $\square$

It may be interesting to note the following facts. Suppose that $X = X_1 \cap X_2$, where X_1 is convex and X_2 is polyhedral. If the set X_2 satisfies one of the conditions (1) or (2) or, more generally, if its lineality space is $\{0\}$, then it is obvious that $\text{Ker}(\nabla h(x^*)^T) \cap \text{Ker}(\nabla g(x^*)^T) = \{0\}$ (g and h are the constraints defining X) so that

condition (b) in Theorem 4.3 becomes superfluous. In turn, the previous observation can be applied in the following way. Suppose that we have a $\text{VIP}(X, F)$ with X (described by general constraints) contained, for example, in the nonnegative orthant. In order to guarantee that a stationary point is a global minimum of Ψ we need conditions (a) and (b) of Theorem 4.3. However, we can consider $\text{VIP}(F, X \cap \{x | x \geq 0\})$. This latter problem is obviously equivalent to the original one. Nevertheless, as observed before, requirement (b) of Theorem 4.3 is now superfluous.

In [10] we considered the case, often encountered in applications, in which X is a rectangle. Then, weaker conditions can be obtained. In particular we showed in [10] that if $X = \mathbb{R}_+^n$, i.e., if the VIP reduces to a complementarity problem, then F being a P_0 -function is a sufficient condition for guaranteeing that every stationary point of Ψ is a global solution.

REFERENCES

- [1] A. BACHEM AND W. KERN, *Linear Programming Duality. An Introduction to Programming Duality*, Springer-Verlag, Heidelberg, 1992.
- [2] M. CAO AND M. C. FERRIS, *A pivotal method for affine variational inequalities*, Math. Oper. Res., 21 (1996), pp. 44–64.
- [3] F. H. CLARKE, *Optimization and Nonsmooth Analysis*, John Wiley, New York, 1983 (reprinted by SIAM, Philadelphia, 1990).
- [4] R. W. COTTLE, J.-S. PANG AND R. E. STONE, *The Linear Complementarity Problem*, Academic Press, Boston, 1992.
- [5] T. DE LUCA, F. FACCHINEI, AND C. KANZOW, *A semismooth equation approach to the solution of nonlinear complementarity problems*, Math. Programming, 75 (1996), pp. 407–439.
- [6] S. P. DIRKSE AND M. C. FERRIS, *The PATH solver: A non-monotone stabilization scheme for mixed complementarity problems*, Optimization Methods and Software, 5 (1995), pp. 123–156.
- [7] B. C. EAVES, *On the basic theorem of complementarity*, Math. Programming, 1 (1971), pp. 68–75.
- [8] F. FACCHINEI AND C. KANZOW, *A nonsmooth inexact Newton method for the solution of large-scale nonlinear complementarity problems*, Math. Programming, 76 (1997), pp. 493–512.
- [9] F. FACCHINEI, A. FISCHER, AND C. KANZOW, *Inexact Newton methods for semismooth equations with applications to variational inequality problems*, in Nonlinear Optimization and Applications, G. Di Pillo and F. Giannessi, eds., Plenum Press, New York, 1996, pp. 125–139.
- [10] F. FACCHINEI, A. FISCHER, AND C. KANZOW, *A semismooth Newton method for variational inequalities: The case of box constraints*, in Complementarity and Variational Problems: State of the Art, M. C. Ferris and J.-S. Pang, eds., SIAM, Philadelphia, 1997, pp. 76–90.
- [11] F. FACCHINEI, A. FISCHER, AND C. KANZOW, *A semismooth Newton method for variational inequalities: Theoretical results and preliminary numerical results*, Technical Report 102, Institute of Applied Mathematics, University of Hamburg, Hamburg, Germany, 1995.
- [12] F. FACCHINEI AND J. SOARES, *A new merit function for nonlinear complementarity problems and a related algorithm*, SIAM J. Optim., 7 (1997), pp. 225–247.
- [13] M. C. FERRIS AND J.-S. PANG, *Engineering and economic applications of complementarity problems*, SIAM Rev., to appear.
- [14] A. FISCHER, *A special Newton-type optimization method*, Optimization, 24 (1992), pp. 269–284.
- [15] A. FISCHER, *An NCP-function and its use for the solution of complementarity problems*, in Recent Advances in Nonsmooth Optimization, D. Z. Du, L. Qi, and R. S. Womersley, eds., World Scientific, Singapore, 1995, pp. 88–105.
- [16] M. FUKUSHIMA, *Merit functions for variational inequality and complementarity problems*, in Nonlinear Optimization and Applications, G. Di Pillo and F. Giannessi, eds., Plenum Press, New York, 1996, pp. 155–170.
- [17] P. T. HARKER AND J.-S. PANG, *Finite-dimensional variational inequality and nonlinear complementarity problems: a survey of theory, algorithms and applications*, Math. Programming, 48 (1990), pp. 161–220.
- [18] H. JIANG, *Local properties of solutions of nonsmooth variational inequalities*, Optimization, 33 (1995), pp. 119–132.

- [19] A. J. KING AND R. T. ROCKAFELLAR, *Sensitivity analysis for nonsmooth generalized equations*, Math. Programming, 55 (1992), pp. 193–212.
- [20] M. KOJIMA, *Strongly stable stationary solutions in nonlinear programs*, in Analysis and Computation of Fixed Points, S. M. Robinson, ed., Academic Press, New York, 1980, pp. 93–138.
- [21] J. LIU, *Strong stability in variational inequalities*, SIAM J. Control Optim., 33 (1995), pp. 725–749.
- [22] A. NAGURNEY, *Network Economics: A Variational Inequality Approach*, Kluwer Academic Publishers, Boston, 1993.
- [23] J.-S. PANG, *Complementarity problems*, in Handbook of Global Optimization, R. Horst and P. M. Pardalos, eds., Kluwer Academic Publishers, Dordrecht, 1995, pp. 271–338.
- [24] J.-S. PANG AND L. QI, *Nonsmooth equations: motivation and algorithms*, SIAM J. Optim., 3 (1993), pp. 443–465.
- [25] L. QI AND J. SUN, *A nonsmooth version of Newton's method*, Math. Programming, 58 (1993), pp. 353–367.
- [26] D. RALPH, *Global convergence of damped Newton's method for nonsmooth equations via the path search*, Math. Oper. Res., 19 (1994), pp. 352–389.
- [27] S. M. ROBINSON, *Generalized equations and their solutions. Part I: Basic theory*, Math. Programming Study, 10 (1979), pp. 128–141.
- [28] S. M. ROBINSON, *Strongly regular generalized equations*, Math. Oper. Res., 5 (1980), pp. 43–62.
- [29] S. M. ROBINSON, *Normal maps induced by linear transformations*, Math. Oper. Res., 17 (1992), pp. 691–714.
- [30] S. M. ROBINSON, *Homeomorphism conditions for normal maps of polyhedra*, in Optimization and Nonlinear Analysis, A. Ioffe, M. Marcus, and S. Reich, eds., Longman Scientific and Technical, Harlow, U.K., 1992, pp. 240–248.
- [31] S. M. ROBINSON, *Nonsingularity and symmetry for linear normal maps*, Math. Programming, 62 (1993), pp. 415–425.
- [32] S. M. ROBINSON, *Sensitivity analysis of variational inequalities by normal-map techniques*, in Nonlinear Variational Inequalities and Network Equilibrium Problems, F. Giannessi and A. Maugeri, eds., Plenum Press, New York, 1995, pp. 257–269.
- [33] H. SELLAMI AND S. M. ROBINSON, *Homotopies based on nonsmooth equations for solving nonlinear variational inequalities*, in Nonlinear Optimization and Applications, G. Di Pillo and F. Giannessi, eds., Plenum Press, New York, 1996, pp. 329–343.
- [34] B. XIAO AND P. T. HARKER, *A nonsmooth Newton method for variational inequalities, parts I and II*, Math. Programming, 65 (1994), pp. 151–216.

INTERIOR POINT TRAJECTORIES IN SEMIDEFINITE PROGRAMMING*

D. GOLDFARB[†] AND K. SCHEINBERG[‡]

Abstract. In this paper we study interior point trajectories in semidefinite programming (SDP) including the central path of an SDP. This work was inspired by the seminal work of Megiddo on linear programming trajectories [*Progress in Math. Programming: Interior-Point Algorithms and Related Methods*, N. Megiddo, ed., Springer-Verlag, Berlin, 1989, pp. 131–158]. Under an assumption of primal and dual strict feasibility, we show that the primal and dual central paths exist and converge to the analytic centers of the optimal faces of, respectively, the primal and the dual problems. We consider a class of trajectories that are similar to the central path but can be constructed to pass through any given interior feasible or infeasible point, and study their convergence. Finally, we study the derivatives of these trajectories and their convergence.

Key words. semidefinite programming, central path, interior point methods

AMS subject classifications. 90C25, 49M05, 65K05

PII. S105262349630009X

1. Introduction. The purpose of this paper is to study properties of the trajectories associated with interior point methods for semidefinite programming (SDP) problems. Since many aspects of semidefinite programming find close analogues in linear programming, several interior point methods designed for linear programming (LP) have been successfully applied to SDP (e.g., see [2], [4], [11], [14], [19], [21], [22], [23], [29], [32]). Many of these aspects have also been studied in the more general framework of self-scaled cones in [22], [23].

Many interior point methods can be viewed as iterative approximations to continuous path-following methods. Our aim is to provide a theoretical basis for such methods for SDP by describing the limiting behavior of the continuous central path and related trajectories for such problems.

This work is an extension of the linear programming results in [17] to SDP. We characterize the optimal face of an SDP problem and prove that the central path converges to the analytic center of the optimal face. Unlike LP problems, an SDP problem does not always have a strictly complementary primal-dual pair of solutions (e.g., see [3], [14]). Thus the SDP central path cannot be guaranteed to converge to such a pair as it does in LP. However, we show that it converges to a “least nonstrictly complementary” pair in the sense that the sum of the ranks of the primal and dual solutions (viewed as matrices) is as large as possible.

Since in practice it is difficult to obtain a point on the central path, it is important to consider trajectories that have properties similar to those of the central path but which pass through a given feasible or infeasible interior (i.e., positive definite) primal solution. For LP, trajectories that pass through a given feasible interior primal solution were introduced in [6] and [1] and are called primal affine scaling (PAS)

*Received by the editors March 7, 1996; accepted for publication (in revised form) April 21, 1997; published electronically August 26, 1998. This research was supported in part by NSF grants DMS 91-06195, DMS 94-14438, and DMS 95-27124 and by DOE grant DE-FG02-92ER25126. The research of the second author was supported in part by an IBM Cooperative Fellowship.

<http://www.siam.org/journals/siopt/8-4/30009.html>

[†]Department of Industrial Engineering and Operations Research, Columbia University, New York, NY 10027-6699 (goldfarb@columbia.edu).

[‡]IBM T.J. Watson Research Center, Yorktown Heights, NY 10598 (katyas@watson.ibm.com).

trajectories, due to the fact that they correspond to continuous versions of primal affine scaling iterative algorithms. An extension of this class of trajectories to those that pass through given infeasible and feasible interior primal and dual solutions has been studied for the case of monotone linear and nonlinear complementarity problems in [13] and [10]. We study such trajectories, which we shall refer to as “infeasible” central paths, for SDP problems and extend our convergence results to them. They are important because of their connection with primal-dual infeasible interior point methods for SDP (e.g., see [14] and [25]).

We show that under the assumptions of primal and dual nondegeneracy and strict complementarity defined in [3], the first order derivatives of the central path are bounded in the limit. We also provide formulae for the limit of these derivatives and show that the factorization of only one matrix is required to compute these and all higher order derivatives of a solution on the central path.

The paper is organized as follows. In section 2 we describe the central path for a primal-dual pair of SDP problems and introduce our basic assumptions and some notation. In section 3 we characterize the optimal faces of the primal and dual SDP problems and prove our main convergence result for the primal-dual central path in section 4. We extend the results of section 4 to the infeasible central path in section 5. Finally, in section 6 we analyze the limiting properties of the derivatives of the central path and show that computation of the derivatives requires factorizing a single matrix for all orders of the derivatives.

2. The central path. In this paper we consider the SDP problem, henceforth referred to as the primal problem

$$(P) \quad \begin{aligned} & \min \quad C \bullet X \\ & \text{s.t.} \quad A_i \bullet X = b_i, \quad i = 1, \dots, m, \\ & \quad \quad X \succeq 0, \quad X \in \mathbf{S}^{n \times n}, \end{aligned}$$

where $C \in \mathbf{S}^{n \times n}$, $A_i \in \mathbf{S}^{n \times n}$, $i = 1, \dots, m$, $\mathbf{S}^{n \times n}$ denotes the space of real symmetric $n \times n$ matrices, and $b \in \mathbf{R}^m$. The inner product on $\mathbf{S}^{n \times n}$ is $A \bullet B = \text{trace}(AB) = \sum_{i,j} A_{ij}B_{ij}$, and by $X \succeq 0$ ($X \succ 0$) we mean that X is positive semidefinite (positive definite).

The problem dual to (P) is the semidefinite programming problem

$$(D) \quad \begin{aligned} & \max \quad \sum_{i=1}^m y_i b_i \\ & \text{s.t.} \quad \sum_{i=1}^m y_i A_i + Z = C, \\ & \quad \quad Z \succeq 0, \quad Z \in \mathbf{S}^{n \times n}. \end{aligned}$$

Throughout the paper the following are assumed to hold.

ASSUMPTION 2.1. *The matrices A_i , $i = 1, \dots, m$, are linearly independent, i.e., $\sum_{i=1}^m u_i A_i = 0$ implies that $u_i = 0$, $i = 1, \dots, m$.*

ASSUMPTION 2.2. *Both the primal and the dual problems have interior feasible solutions, i.e.,*

$$\exists X_0 \in \mathbf{S}^{n \times n} : X_0 \succ 0 \text{ and } A_i \bullet X_0 = b_i, \quad i = 1, \dots, m,$$

and

$$\exists(y_0, Z_0) \in \mathbf{R}^m \times \mathbf{S}^{n \times n} : Z_0 \succ 0 \text{ and } \sum_{i=1}^m y_{0i} A_i + Z_0 = C.$$

Under Assumption 2.2 both primal and dual problems have finite optimal solutions, $\bar{X}$ and $(\bar{y}, \bar{Z})$, and the duality gap $\bar{X} \bullet \bar{Z} = 0$ [21]. The optimal solutions also satisfy $\bar{X} \bar{Z} = \bar{Z} \bar{X} = 0$ [30].

The central path for the problem (P) is a trajectory of the solutions $X(\mu) \in \mathbf{S}^{n \times n}$ to the following parametric family of problems for values of the parameter $\mu > 0$ [15], [21], [29]:

$$\begin{aligned} \min \quad & C \bullet X - \mu(\ln \det X) \\ \text{(P}_\mu\text{)} \quad & \text{s.t. } A_i \bullet X = b_i, \quad i = 1, \dots, m, \\ & X \succ 0, \quad X \in \mathbf{S}^{n \times n}. \end{aligned}$$

From Assumption 2.2 and the strict convexity of the logarithmic barrier objective function for any $\mu > 0$, problem (P_μ) has a unique solution that satisfies the Karush–Kuhn–Tucker optimality conditions for (P_μ):

$$\begin{aligned} \text{(CP}_\mu\text{)} \quad & Z = \mu X^{-1}, \\ & A_i \bullet X = b_i, \quad i = 1, \dots, m, \\ & \sum_{i=1}^m y_i A_i + Z = C, \\ & X, Z \succ 0, \quad X, Z \in \mathbf{S}^{n \times n}. \end{aligned}$$

The central path for the dual problem can be defined in an analogous manner and is the trajectory $(y(\mu), Z(\mu)) \in \mathbf{R}^m \times \mathbf{S}^{n \times n}$ whose points satisfy the same system (CP_μ) as the points $X(\mu)$ on the primal central path. Hence it makes sense to refer to the trajectory $(X(\mu), y(\mu), Z(\mu))$, $\mu > 0$, of solutions to (CP_μ) as the primal-dual central path. Under Assumption 2.2, not only does this path exist but also it converges to an optimal primal-dual solution (e.g., see [15], [21], [30]).

To conclude this section we introduce some notation that we will use later in the paper. First, we note that the variables X and Z can be viewed both as symmetric matrices and as vectors (obtained from these matrices by stacking their columns one after the other) lying in a $n(n+1)/2$ -dimensional subspace of $\mathbf{R}^{n^2}$. Whenever we refer to the matrix X as a vector, we denote it by $\text{vec}(X)$. By the constraint matrix A we denote the $m \times n^2$ matrix, the i th row of which equals $\text{vec}(A_i)^T$. Note that $C \bullet X = \text{vec}(C)^T \text{vec}(X)$ is the usual inner product.

The Kronecker product $M \otimes N$ of matrices $M \in \mathbf{R}^{n \times n}$ and $N \in \mathbf{R}^{n \times n}$ is defined as

$$M \otimes N = \begin{bmatrix} M_{11}N & \dots & M_{1n}N \\ \vdots & & \vdots \\ M_{n1}N & \dots & M_{nn}N \end{bmatrix}.$$

There are two properties of the Kronecker product that we will need later: $\text{vec}(MNP) = (P^T \otimes M) \text{vec}(N)$ and $(M \otimes P)(N \otimes S) = (MN) \otimes (PS)$ [9].

Last, from properties of the trace we have the following.

PROPERTY 2.3. *Let $A \in \mathbf{R}^{n \times n}$, $X \in \mathbf{R}^{r \times r}$, and $P \in \mathbf{R}^{n \times r}$. Then $A \bullet PXP^T = P^TAP \bullet X$.*

PROPERTY 2.4. *Let $A \in \mathbf{R}^{n \times n}$, $A \succeq 0$, and $B \in \mathbf{R}^{n \times n}$, $B \succeq 0$. Then $A \bullet B \geq 0$.*

3. Optimal faces of the primal and dual problems. Properties of the faces of the cone of positive semidefinite matrices are studied in [5]. The facial structure of SDP problems (i.e., the intersection of the cone of positive semidefinite matrices with an affine subspace) is studied in general terms in [24], [26]. Here we derive a particular system that describes the optimal face of an SDP problem. First, let us introduce some more notation and recall some well-known facts.

If X is a positive semidefinite symmetric matrix, then X has a spectral factorization $X = Q\Lambda Q^T$, where Q is an orthogonal matrix of eigenvectors of X and Λ is a diagonal matrix whose diagonal elements are the nonnegative eigenvalues of X . X can also be represented as

$$X = Q_P \Lambda_P Q_P^T, \quad \Lambda_P \succ 0,$$

where Λ_P is a diagonal matrix whose diagonal elements are the positive eigenvalues of X and Q_P is a matrix with orthonormal columns that are eigenvectors corresponding to these eigenvalues. Throughout this paper the uppercase letter Q will always denote a matrix with orthonormal columns, and Λ and Ω will always denote diagonal matrices of eigenvalues. If Q has a subscript, then the matrix it represents will, in general, not be square. Let $\mathcal{R}(X)$ denote the range (column space) of X . Clearly, $\mathcal{R}(X) = \text{span}(Q_P)$, the subspace spanned by the columns of Q_P , and the dimension of this subspace (i.e., the number of positive eigenvalues of X) equals the rank of X . If $\bar{X}$ and $\bar{Z}$ are an optimal primal-dual pair of solutions, where $\bar{X} = Q_P \Lambda_P Q_P^T$, $\Lambda_P \succ 0$ and $\bar{Z} = Q_D \Omega_D Q_D^T$, $\Omega_D \succ 0$, then $Q_P^T Q_D = 0$ follows from the complementarity of $\bar{X}$ and $\bar{Z}$. Hence $\mathcal{R}(\bar{X}) \subseteq \text{Ker}(\bar{Z})$ and $\mathcal{R}(\bar{Z}) \subseteq \text{Ker}(\bar{X})$.

Let $\mathbf{O}_P$ denote the primal optimal face, i.e., the set of primal optimal solutions, and let $\mathbf{O}_D$ denote the dual optimal face. Note that both $\mathbf{O}_P$ and $\mathbf{O}_D$ are convex subsets of affine subspaces of $\mathbf{S}^{n \times n}$. By $\text{ri } \mathbf{O}_P$ ($\text{ri } \mathbf{O}_D$) we denote the relative interior of $\mathbf{O}_P$ ($\mathbf{O}_D$). Then the following lemma holds [5].

LEMMA 3.1. *For any $\bar{X} \in \mathbf{O}_P$ ($(\bar{y}, \bar{Z}) \in \mathbf{O}_D$) and any $\tilde{X} \in \text{ri } \mathbf{O}_P$ ($(\tilde{y}, \tilde{Z}) \in \text{ri } \mathbf{O}_D$), $\mathcal{R}(\bar{X}) \subseteq \mathcal{R}(\tilde{X})$ ($\mathcal{R}(\bar{Z}) \subseteq \mathcal{R}(\tilde{Z})$).*

This lemma shows that if both $\bar{X}$ and $\tilde{X}$ are in $\text{ri } \mathbf{O}_P$, then $\mathcal{R}(\bar{X}) = \mathcal{R}(\tilde{X})$ and hence shows that any $\tilde{X} \in \text{ri } \mathbf{O}_P$ is an optimal solution of maximum rank. Let us denote the subspace $\mathcal{R}(\tilde{X})$ by $\mathbf{R}_P$. Analogously, let $\mathbf{R}_D$ be the subspace spanned by the eigenvectors corresponding to the positive eigenvalues of $\tilde{Z}$ for any dual solution $(\tilde{y}, \tilde{Z})$ in the relative interior of $\mathbf{O}_D$.

Let $\dim \mathbf{R}_P = r$ and $\dim \mathbf{R}_D = s$. From the complementarity of any primal-dual pair of optimal solutions, $\mathbf{R}_P \perp \mathbf{R}_D$. Hence $r + s \leq n$. If $r + s < n$, we say that the primal-dual pair of problems does not satisfy strict complementarity. Note that this can never happen in linear programming.

If we define $\mathbf{R}_N = [\mathbf{R}_P \oplus \mathbf{R}_D]^\perp$ ($\mathbf{R}_N = \{0\}$ if $r + s = n$), then $\mathbf{R}_P \perp \mathbf{R}_N \perp \mathbf{R}_D$ and $\mathbf{R}_P \oplus \mathbf{R}_N \oplus \mathbf{R}_D = \mathbf{R}^n$; i.e., we have a partition of $\mathbf{R}^n$ into three mutually orthogonal subspaces.

Let Q_P be any $n \times r$ matrix whose columns form an orthonormal basis for $\mathbf{R}_P$. Then any solution $X \in \mathbf{O}_P$ can be written as $X = Q_P U Q_P^T$, $U \succeq 0$, so the optimal face of (P) is given by the set of the solutions to the following system:

$$(1) \quad \begin{aligned} A_i \bullet Q_P U Q_P^T &= b_i, \quad i = 1, \dots, m, \\ U &\succeq 0, \quad U \in \mathbf{S}^{r \times r}. \end{aligned}$$

Indeed, for any U feasible for (1), $Q_P U Q_P^T$ is feasible for (P), and since $Q_P U Q_P^T \bullet Z = 0$,

for any $Z \in \mathbf{O}_D$ (from $\mathbf{R}_P \perp \mathbf{R}_D$), $Q_P U Q_P^T$ and Z satisfy complementary slackness. Therefore, $Q_P U Q_P^T$ is an optimal solution to (P).

Similarly, let the columns of Q_D form an orthonormal basis for $\mathbf{R}_D$. Then any optimal dual solution can be written as $Z = Q_D V Q_D^T$, $V \succeq 0$, and the optimal face of (D) is given by the set of solutions to the system

$$(2) \quad \sum_{i=1}^m y_i A_i + Q_D V Q_D^T = C, \\ V \succeq 0, \quad V \in \mathbf{S}^{s \times s}.$$

Notice that the definitions of the primal and dual optimal faces are invariant with respect to the choices of Q_P and Q_D as long as their columns form orthonormal bases for the subspaces $\mathbf{R}_P$ and $\mathbf{R}_D$, respectively.

The following lemma shows that under Assumptions 2.1 and 2.2 both primal and dual optimal faces are bounded. Thus their analytic centers are well defined, which is important for the results of the next section.

LEMMA 3.2. *Under Assumptions 2.1 and 2.2 the optimal sets of the primal and the dual problems are bounded.*

Proof. Suppose that the set of optimal dual solutions is unbounded. That is, there exists a nonzero direction (u, W) satisfying

$$\sum_{i=1}^m u_i b_i = 0, \\ \sum_{i=1}^m u_i A_i + W = 0, \\ W \succeq 0.$$

Multiplying the second equation by an interior feasible primal solution X_0 , which exists by Assumption 2.2, we obtain

$$0 = \sum_{i=1}^m u_i A_i \bullet X_0 + W \bullet X_0 = \sum_{i=1}^m u_i b_i + W \bullet X_0 = W \bullet X_0.$$

Therefore, $W = 0$ since $W \bullet X_0 = \text{trace}(W X_0) = \text{trace}(W^{1/2} X_0 W^{1/2})$ and $X_0 \succ 0$. It then follows that $\sum_i u_i A_i = 0$, which by Assumption 2.1 implies that $u = 0$.

The boundedness of the set of optimal primal solutions can be proved in a similar manner. $\square$

4. Convergence of the central path. We prove in this section that the primal central path converges to the analytic center of the optimal face $\mathbf{O}_P$. First we show that the limit $\bar{X}$ of the central path is in the relative interior of the optimal face. Then we show that $\bar{X}$ is, in fact, the analytic center of the optimal face. We then extend these results to the dual central path.

In [31] it is shown that in the case of convex homogeneous self-dual cones, which includes the case of the cone of positive semidefinite matrices, the central path converges to a strictly complementary solution provided that one exists. In [16], under the assumption of strict complementarity, it is shown that the primal-dual central path of an SDP problem converges to the analytic center of the optimal face. We obtain the same results without assuming strict complementarity.

Let $\bar{X}$ be the limit of the primal central path as $\mu \rightarrow 0$.

LEMMA 4.1. *There exists a spectral factorization $\bar{X} = \bar{Q}\bar{\Lambda}\bar{Q}^T$ and a sequence $\{\mu_k\}$ such that $X(\mu_k) \rightarrow \bar{X}$, $Q(\mu_k) \rightarrow \bar{Q}$, and $\Lambda(\mu_k) \rightarrow \bar{\Lambda}$, where $Q(\mu_k)\Lambda(\mu_k)Q^T(\mu_k)$ is a spectral factorization of $X(\mu_k)$.*

Proof. The proof follows trivially from the compactness of the set of the orthogonal matrices. Notice that the limit $\bar{\Lambda}$ is uniquely defined by $\bar{X}$, and the limit $\bar{Q}$, generally speaking, depends on the sequence $\{\mu_k\}$. $\square$

We know that $\bar{X}$ and $(\bar{y}, \bar{Z})$ are optimal solutions to the primal and dual problems, respectively. We first want to prove that each is in the relative interior of the optimal face for its respective problem.

LEMMA 4.2. *$\bar{X}$ ($(\bar{y}, \bar{Z})$) belongs to the relative interior of the primal (dual) optimal face.*

Proof. Let $(X(\mu), y(\mu), Z(\mu))$ be a point on the central path. Let $\tilde{X} \in \text{ri}\mathbf{O}_P$ and $(\tilde{y}, \tilde{Z}) \in \text{ri}\mathbf{O}_D$. It is trivial to verify that

$$[\tilde{X} - X(\mu)] \bullet [\tilde{Z} - Z(\mu)] = 0,$$

and since $\tilde{X} \bullet \tilde{Z} = 0$ and $X(\mu)Z(\mu) = \mu I$, we obtain

$$(3) \quad \tilde{X} \bullet X(\mu)^{-1} + \tilde{Z} \bullet Z(\mu)^{-1} = n.$$

Now both terms on the left side of (3) are nonnegative by Proposition 2.4; hence

$$(4) \quad \tilde{X} \bullet X(\mu)^{-1} \leq n.$$

Consider the sequence $\{\mu_k\}$ as defined in Lemma 4.1, such that $X(\mu_k) \rightarrow \bar{X}$ and the spectral factorizations $Q(\mu_k)\Lambda(\mu_k)Q^T(\mu_k)$ of $X(\mu_k)$ converge. $\bar{X} = \bar{Q}\bar{\Lambda}\bar{Q}^T$ and $\bar{X} = \bar{Q}_P\bar{\Lambda}_P\bar{Q}_P^T$, $\bar{\Lambda}_P \succ 0$, $\bar{\Lambda}_P \in \mathbf{S}^{r \times r}$. (The columns of $\bar{Q}_P$ are eigenvectors of $\bar{X}$ that span $\mathbf{R}_P$.) Let us order the columns of $\bar{Q}$ and partition $\bar{Q}$ into two parts $[\bar{Q}_P, \bar{Q}_{ND}]$ so that $\bar{Q}_P$ has r columns and $\bar{Q}_P^T\bar{Q}_P$ is nonsingular. This is always possible since $\bar{Q}^T\bar{Q}_P$ has full column rank. Let us order the columns of $Q(\mu_k)$ and the columns and rows of Λ and $\Lambda(\mu_k)$ and partition them according to the column order and partitioning of $\bar{Q}$. Then $\bar{X} = \bar{Q}_P\bar{\Lambda}_P\bar{Q}_P^T + \bar{Q}_{ND}\bar{\Lambda}_{ND}\bar{Q}_{ND}^T$ and $X^{-1}(\mu_k) = Q_P(\mu_k)\Lambda_P^{-1}(\mu_k)Q_P^T(\mu_k) + Q_{ND}(\mu_k)\Lambda_{ND}^{-1}(\mu_k)Q_{ND}^T(\mu_k)$, and from (4),

$$\bar{Q}_P\bar{\Lambda}_P\bar{Q}_P^T \bullet Q_P(\mu_k)\Lambda_P^{-1}(\mu_k)Q_P^T(\mu_k) + \bar{Q}_P\bar{\Lambda}_P\bar{Q}_P^T \bullet Q_{ND}(\mu_k)\Lambda_{ND}^{-1}(\mu_k)Q_{ND}^T(\mu_k) \leq n.$$

Since both terms in this sum are nonnegative by Property 2.4, it follows from Property 2.3 that

$$(5) \quad Q_P^T(\mu_k)\bar{Q}_P\bar{\Lambda}_P\bar{Q}_P^TQ_P(\mu_k) \bullet \Lambda_P^{-1}(\mu_k) = \bar{Q}_P\bar{\Lambda}_P\bar{Q}_P^T \bullet Q_P(\mu_k)\Lambda_P^{-1}(\mu_k)Q_P^T(\mu_k) \leq n.$$

Now, $\lambda_i(\mu_k) \rightarrow \bar{\lambda}_i$ and $Q_P^T(\mu_k)\bar{Q}_P\bar{\Lambda}_P\bar{Q}_P^TQ_P(\mu_k) \rightarrow \bar{Q}_P^T\bar{Q}_P\bar{\Lambda}_P\bar{Q}_P^T\bar{Q}_P \equiv \bar{U}_P \in \mathbf{S}^{r \times r}$. We have $\bar{\Lambda}_P \succ 0$, and $\bar{Q}_P^T\bar{Q}_P$ is nonsingular; hence $\bar{U}_P \succ 0$. It then follows from (5) that

$$\sum_{i=1}^r \bar{\lambda}_i^{-1}(\bar{U}_P)_{ii} \leq n,$$

where $(\bar{U}_P)_{ii} > 0$ (because $\bar{U}_P \succ 0$). Since the sum of the ratios $(\bar{U}_P)_{ii}/\bar{\lambda}_i$ is finite, it follows that $\bar{\lambda}_i > 0$, $i = 1, \dots, r$. Therefore, $\bar{X}$ has rank r proving that $\bar{X} \in \text{ri}\mathbf{O}_P$. Similarly, it can be shown that $(\bar{y}, \bar{Z}) \in \text{ri}\mathbf{O}_D$. $\square$

From Lemma 3.2 it follows that the analytic center of the optimal face $\mathbf{O}_P$ is well defined. We now show that $\bar{X}$ is, in fact, this analytic center.

THEOREM 4.3. *Let $\bar{X}$ be the limit of the primal central path as $\mu \rightarrow 0$. Then $\bar{X} = Q_P \bar{U} Q_P^T$, where $\bar{U}$ depends on the choice of the orthonormal basis Q_P for $\mathbf{R}_P$ and is the unique solution to the problem*

$$(6) \quad \begin{aligned} & \max \quad \ln \det U \\ & \text{s.t.} \quad Q_P^T A_i Q_P \bullet U = b_i, \quad i = 1, \dots, m, \\ & \quad \quad U \succ 0, \quad U \in \mathbf{S}^{r \times r}; \end{aligned}$$

i.e., $\bar{X}$ is the analytic center of the primal optimal face.

Proof. Problem (6) can be rewritten in an equivalent form:

$$(7) \quad \begin{aligned} & \max \quad \ln \det U \\ & \text{s.t.} \quad Q_P^T A_i Q_P \bullet U = b_i, \\ & \quad \quad Q_P^T C Q_P \bullet U = c(0), \\ & \quad \quad U \succ 0, \quad U \in \mathbf{S}^{r \times r}, \end{aligned}$$

where $c(0)$ is the optimal objective function value. Since the objective function is strictly concave, the solution of this problem is unique and satisfies the following system of optimality conditions:

$$(8) \quad \begin{aligned} & U^{-1} - \sum y_i Q_P^T A_i Q_P = t Q_P^T C Q_P, \\ & \quad \quad Q_P^T A_i Q_P \bullet U = b_i, \\ & \quad \quad Q_P^T C Q_P \bullet U = c(0). \end{aligned}$$

For any fixed $\mu > 0$, let $C \bullet X(\mu) = c(\mu)$, where $X(\mu)$ is the solution to the optimality conditions (CP_μ) for problem (P_μ) . Then $X(\mu)$ is a solution to the following problem:

$$(9) \quad \begin{aligned} & \max \quad \ln \det X \\ & \text{s.t.} \quad A_i \bullet X = b_i, \quad i = 1, \dots, m, \\ & \quad \quad C \bullet X = c(\mu), \\ & \quad \quad X \succ 0, \quad X \in \mathbf{S}^{n \times n}. \end{aligned}$$

Using notation of Lemma 4.2, $\bar{X} = \bar{Q} \bar{\Lambda} \bar{Q}^T = \bar{Q}_P \bar{\Lambda}_P \bar{Q}_P^T + \bar{Q}_{ND} \bar{\Lambda}_{ND} \bar{Q}_{ND}^T$, where $\bar{Q} = [\bar{Q}_P \bar{Q}_{ND}]$ is a matrix of eigenvectors of $\bar{X}$ and $\bar{\Lambda}_P$ is a diagonal matrix of positive eigenvalues of $\bar{X}$ (r of them) and $\bar{\Lambda}_{ND} = 0$. From Lemma 4.2 it follows that $\bar{Q}_P$ spans $\mathbf{R}_P$.

As in Lemma 4.2, consider the convergent sequence $X(\mu_k) = Q(\mu_k) \Lambda(\mu_k) Q^T(\mu_k) = Q_P(\mu_k) \Lambda_P(\mu_k) Q_P^T(\mu_k) + Q_{ND}(\mu_k) \Lambda_{ND}(\mu_k) Q_{ND}^T(\mu_k)$, such that $Q(\mu_k)$ converges to $\bar{Q}$ and $\Lambda(\mu_k)$ converges to $\bar{\Lambda}$ as $\mu_k \rightarrow 0$. Since $X(\mu_k) = Q(\mu_k) \Lambda(\mu_k) Q^T(\mu_k)$ and $\Lambda(\mu_k)$ is diagonal, requiring X in (9) to be of the form

$$\begin{aligned} X &= Q(\mu_k) \begin{bmatrix} U & 0 \\ 0 & V \end{bmatrix} Q^T(\mu_k) = Q_P(\mu_k) U Q_P^T(\mu_k) + Q_{ND}(\mu_k) V Q_{ND}^T(\mu_k), \\ & \quad \quad U \in \mathbf{S}^{r \times r}, \quad V \in \mathbf{S}^{n-r \times n-r}, \end{aligned}$$

where $U \succ 0$ and V is equal to $\Lambda_{ND}(\mu_k)$, does not affect the solution of problem (9). We restrict V and not U because, as we have already shown, the sequence of the solutions $X = X(\mu)$ to (9) converges to an optimal solution, where $V = 0$ and U is a positive definite matrix. Also, $\ln \det X = \ln \det U + \ln \det V$. Therefore, from Property 2.3, (9) is equivalent to

$$(10) \quad \begin{aligned} & \max \quad \ln \det U \\ & \text{s.t.} \quad Q_P^T(\mu_k) A_i Q_P(\mu_k) \bullet U = b_i - Q_{ND}^T(\mu_k) A_i Q_{ND}(\mu_k) \bullet \Lambda_{ND}(\mu_k), \\ & \quad Q_P^T(\mu_k) C Q_P(\mu_k) \bullet U = \bar{c}(\mu_k) - Q_{ND}^T(\mu_k) C Q_{ND}(\mu_k) \bullet \Lambda_{ND}(\mu_k), \\ & \quad U \succ 0, \quad U \in S^{r \times r}. \end{aligned}$$

The unique optimal solution $U = \Lambda_P(\mu_k)$ of (10) satisfies the following system:

$$(11) \quad \begin{aligned} & U^{-1} - \sum y_i(\mu_k) Q_P(\mu_k) A_i Q_P^T(\mu_k) = t(\mu_k) Q_P(\mu_k) C Q_P^T(\mu_k), \\ & Q_P^T(\mu_k) A_i Q_P(\mu_k) \bullet U = b_i - Q_{ND}^T(\mu_k) A_i Q_{ND}(\mu_k) \bullet \Lambda_{ND}(\mu_k), \\ & Q_P^T(\mu_k) C Q_P(\mu_k) \bullet U = \bar{c}(\mu_k) - Q_{ND}^T(\mu_k) C Q_{ND}(\mu_k) \bullet \Lambda_{ND}(\mu_k), \\ & U \succ 0. \end{aligned}$$

As $\mu_k \rightarrow 0$, $Q(\mu_k) \rightarrow \bar{Q}$ and the system (11) converges to (8) with $Q_P = \bar{Q}_P$. Since $\Lambda_P(\mu_k)$ is the solution to (11), then the limit $\bar{\Lambda}_P \succ 0$ has to satisfy (8). This proves that $\bar{X}$ is the analytic center of the primal optimal face. $\square$

As in LP, problems (P) and (D) can be written in a “symmetric” form. Specifically, we can use the “conic” formulation given in Chapter 3 of [21] (see also [29]). Let $\mathbf{L}$ be the subspace of $\mathbf{S}^{n \times n}$ spanned by A_i , $i = 1, \dots, m$, and $D \in \mathbf{S}^{n \times n}$ be such that $A_i \bullet D = b_i$, $i = 1, \dots, m$. Then problems (P) and (D) can be formulated as

$$(P') \quad \begin{aligned} & \min \quad C \bullet X \\ & \text{s.t.} \quad X \in \{D + \mathbf{L}^\perp\}, \\ & \quad X \succeq 0, \quad X \in \mathbf{S}^{n \times n}, \end{aligned}$$

and

$$(D') \quad \begin{aligned} & \min \quad D \bullet Z \\ & \text{s.t.} \quad Z \in \{C + \mathbf{L}\}, \\ & \quad Z \succeq 0, \quad Z \in \mathbf{S}^{n \times n}. \end{aligned}$$

Consequently, all the results in this section extend to the dual problem, and in particular, in terms of formulation (D) of the dual, we have the following.

THEOREM 4.4. *Let $(\bar{y}, \bar{Z})$ be a limit point of the dual central path. Then $\bar{Z} = Q_D \bar{W} Q_D^T$, where $\bar{W}$ depends on the choice of the orthonormal basis Q_D for $\mathbf{R}_D$ and is the unique solution to the problem*

$$(12) \quad \begin{aligned} & \max \quad \ln \det W \\ & \text{s.t.} \quad \sum_{i=1}^m y_i A_i + Q_D W Q_D^T = C, \\ & \quad W \succ 0, \quad W \in \mathbf{S}^{s \times s}; \end{aligned}$$

i.e., $(\bar{y}, \bar{Z})$ is the analytic center of the dual optimal face.

5. Infeasible central paths. In this section we study the limiting behavior of infeasible central paths associated with infeasible interior point methods for SDP (e.g., see [14], [25]). The limiting behavior of the infeasible central path for linear and nonlinear monotone complementarity problems is studied in [13] and [10], and the existence and convergence of the infeasible central path for monotone semidefinite complementarity problems are studied in [28].

Consider the following perturbation of the primal SDP problem (P):

$$\begin{aligned}
 (\text{IP}_\mu) \quad & \min \quad C \bullet X - \mu(T \bullet X + \ln \det X) \\
 & \text{s.t.} \quad A_i \bullet X = b_i - \mu d_i, \quad i = 1, \dots, m, \\
 & \quad X \succ 0,
 \end{aligned}$$

where $T \in \mathbf{S}^{n \times n}$ is some arbitrary fixed symmetric matrix and $d \in \mathbf{R}^m$ is some arbitrary vector.

If problem (IP_μ) has a solution $X(\mu)$ for some $\mu > 0$, then this solution is unique and satisfies the Karush–Kuhn–Tucker necessary conditions

$$\begin{aligned}
 (\text{ICP}_\mu) \quad & Z = \mu X^{-1}, \\
 & A_i \bullet X = b_i - \mu d_i, \quad i = 1, \dots, m, \\
 & \sum_{i=1}^m y_i A_i + Z = C - \mu T, \\
 & X, Z \succ 0.
 \end{aligned}$$

These conditions are also necessary conditions for $(y(\mu), Z(\mu))$ to be the (unique) optimal solution to the perturbation of the dual SDP problem (D):

$$\begin{aligned}
 (\text{ID}_\mu) \quad & \max \quad y^T b - \mu(y^T d - \ln \det Z), \\
 & \sum_{i=1}^m y_i A_i + Z = C - \mu T, \\
 & Z \succ 0.
 \end{aligned}$$

For some given symmetric matrix $T \in \mathbf{S}^{n \times n}$ and vector $d \in \mathbf{R}^m$, we shall refer to the trajectory of the solutions $(X(\mu), y(\mu), Z(\mu))$ of (ICP_μ) as the infeasible primal-dual central path.

We first discuss the existence of the infeasible central path. For any given $\mu > 0$, $T \in \mathbf{S}^{n \times n}$, and $d \in \mathbf{R}^m$, (ICP_μ) has a unique solution, provided that (ICP_μ) is feasible. It is easy to see that if (ICP_μ) has a solution for some given μ_0 , then it has a solution for any $\mu \in (0, \mu_0)$. Moreover, the following lemma holds.

LEMMA 5.1. *If the feasible sets of problems (P) and (D) are bounded, then for any $T \in \mathbf{S}^{n \times n}$ and any $d \in \mathbf{R}^m$, (ICP_μ) has a unique solution for all $0 < \mu < \infty$.*

This lemma is proved in [27], and a similar result is proved as Theorem 2.4 in [8] for the case of affine scaling trajectories, i.e., $d = 0$. It is also easy to show that the infeasible central path is smooth by applying the implicit function theorem to the system (ICP_μ) for any $\mu > 0$.

The main feature of the infeasible central path is that this path can be constructed to pass through an arbitrary primal infeasible interior solution and an arbitrary dual solution. Specifically, given $X_0 \succ 0$, y_0 , and $\mu_0 > 0$, if we set $Z_0 = \mu_0 X_0^{-1}$,

$$(13) \quad T = \frac{1}{\mu_0} \left[C - \sum_{i=1}^m (y_0)_i A_i - Z_0 \right],$$

and

$$d = \frac{1}{\mu_0} [b_i - A_i \bullet X_0],$$

then (X_0, y_0, Z_0) is the solution to (ICP_μ) for $\mu = \mu_0$. Alternatively, if $Z_0 \succ 0$, y_0 , and $\mu_0 > 0$ are given, then we set $X_0 = \mu_0 Z_0^{-1}$ and define T and d as above. In either case, there is a trajectory $(X(\mu), y(\mu), Z(\mu))$ defined by (ICP_μ) for all $0 < \mu \leq \mu_0$ that passes through the point (X_0, y_0, Z_0) .

We now consider the limiting behavior of the infeasible central path. Let $(\bar{X}, \bar{y}, \bar{Z})$ be a limit point of the solution $(X(\mu), y(\mu), Z(\mu))$ to (ICP_μ) as $\mu \rightarrow 0$. It is obvious that $\bar{X}$ is an optimal solution to (P) and $(\bar{y}, \bar{Z})$ is an optimal solution to (D). As in the case of the central path, we can show that $\bar{X}$ is in the relative interior of the primal optimal face and $(\bar{y}, \bar{Z})$ is in the relative interior of the dual optimal face. The proof is analogous to the proof of Lemma 4.2 with a minor exception. We have

$$\begin{aligned} (X(\mu) - \bar{X}) \bullet (Z(\mu) - \bar{Z}) &= (X(\mu) - \bar{X}) \bullet \sum_{i=1}^m A[y_i(\mu) - \bar{y}_i] \\ &= \sum_{i=1}^m [y_i(\mu) - \bar{y}_i] [b_i - \mu d_i - b_i] = -\mu \sum_{i=1}^m [y_i(\mu) - \bar{y}_i] d_i. \end{aligned}$$

Since $\bar{X} \bullet \bar{Z} = 0$ and $X(\mu) \bullet Z(\mu) = n\mu$, we obtain

$$X(\mu) \bullet \bar{Z} + \bar{X} \bullet Z(\mu) \leq N\mu$$

for some number N large enough, and we can apply the arguments of Lemma 4.2.

More importantly, it is trivial to extend the proof of Theorem 4.3 [27] to give the following.

THEOREM 5.2. *Let $(\bar{X}, \bar{y}, \bar{Z})$ be a limit point of the infeasible central path as $\mu \rightarrow 0$. Then*

1. $\bar{X} = Q_P \bar{U} Q_P^T$, where $\bar{U}$ depends on the choice of the orthonormal basis Q_P for $\mathbf{R}_P$ and is the unique solution to the problem

$$\begin{aligned} (14) \quad & \max \ln \det U + Q_P^T T Q_P \bullet U \\ & \text{s.t. } Q_P^T A_i Q_P \bullet U = b_i, \quad i = 1, \dots, m, \\ & U \succeq 0; \end{aligned}$$

i.e., $\bar{X}$ is the “shifted” analytic center of the primal optimal face.

2. $\bar{Z} = Q_D \bar{W} Q_D^T$, where $\bar{W}$ depends on the choice of the orthonormal basis Q_D for $\mathbf{R}_D$ and is the unique solution to the problem

$$\begin{aligned} (15) \quad & \max \ln \det W - \sum y_i d_i \\ & \text{s.t. } \sum y_i A_i + Q_D W Q_D^T = C, \\ & W \succ 0, \quad W \in \mathbf{S}^{s \times s}; \end{aligned}$$

i.e., $(\bar{y}, \bar{Z})$ is the “shifted” analytic center of the dual optimal face.

Thus the choice of the initial point (X_0, y_0) affects the limit of the trajectory of the primal-dual solutions (obviously, only if the optimal faces are of dimension greater than zero).

We now consider the tangent to the infeasible central path at an arbitrary point on it. Let (X, y, Z) (we omit the argument μ for simplicity) be on the infeasible central path corresponding to given perturbations T and d . Differentiating the system (ICP_μ) with respect to μ for any $\mu > 0$ yields

$$(16) \quad \begin{aligned} \dot{Z} &= -\mu X^{-1} \dot{X} X^{-1} + X^{-1}, \\ A_i \bullet \dot{X} &= -d_i, \quad i = 1, \dots, m, \\ \sum_{i=1}^m \dot{y}_i A_i + \dot{Z} &= -T. \end{aligned}$$

If we multiply the above system by μ and subtract it from (ICP_μ) , we obtain

$$(17) \quad \begin{aligned} Z - \mu \dot{Z} &= \mu^2 X^{-1} \dot{X} X^{-1}, \\ A_i \bullet (X - \mu \dot{X}) &= b_i, \quad i = 1, \dots, m, \\ \sum_{i=1}^m [y_i - \mu \dot{y}_i] A_i + Z - \mu \dot{Z} &= C. \end{aligned}$$

Combining the first and third equations to eliminate $Z - \mu \dot{Z}$, multiplying the result by $X^{1/2}$ on both sides, and noting that $A_i \bullet \dot{X} = X^{1/2} A_i X^{1/2} \bullet X^{-1/2} \dot{X} X^{-1/2}$, we obtain the following equivalent system of equations:

$$(18) \quad \begin{aligned} \bar{A}_i \bullet \Delta &= \bar{A}_i \bullet I - b_i, \quad i = 1, \dots, m, \\ \sum_{i=1}^m \bar{A}_i y_i^E + \mu \Delta &= \bar{C}. \end{aligned}$$

Here $\Delta = \mu X^{-1/2} \dot{X} X^{-1/2}$, $y^E = y - \mu \dot{y}$, $\bar{A}_i = X^{1/2} A_i X^{1/2}$, $i = 1, \dots, m$, and $\bar{C} = X^{1/2} C X^{1/2}$. These equations are the Karush–Kuhn–Tucker conditions with Lagrange multipliers y^E and $\frac{1}{2}\mu$ for the following optimization problem:

$$(19) \quad \begin{aligned} \min \quad & \bar{C} \bullet (I + \Delta) \\ \text{s.t.} \quad & \bar{A}_i \bullet (I + \Delta) = b_i, \quad i = 1, \dots, m, \\ & \Delta \bullet \Delta \leq 1. \end{aligned}$$

When $d = 0$ (i.e., X is primal feasible) the equations in (19) can be replaced by $A_i \bullet \Delta = 0$, $i = 1, \dots, m$, and Δ defines a primal affine scaling direction. Thus, in the general case, Δ can be viewed as a generalized primal affine scaling direction. Note that in vector form $\bar{C} \bullet (I + \Delta) = \text{vec}(C)^T H \text{vec}(I + \Delta)$, the equations in (19) become $AH \text{vec}(I + \Delta) = b$, where $H = X^{1/2} \otimes X^{1/2}$ is the so-called scaling matrix, and $\text{vec}(\dot{X}) = (1/\mu)H \text{vec}(\Delta)$.

If $d = 0$, the infeasible central path is feasible for the primal problem and infeasible for the dual problem. In that case, as we just showed, the trajectories generated are primal affine scaling trajectories. In [1] and [6] for the case of linear programming and in [7] and [8] for the case of SDP, primal affine scaling trajectories are defined by solutions to the following system of Karush–Kuhn–Tucker conditions:

$$\begin{aligned}
(20) \quad & Z = \mu X^{-1} + \mu T, \\
& A_i \bullet X = b_i, \quad i = 1, \dots, m, \\
& \sum_{i=1}^m y_i A_i + Z = C, \\
& X \succ 0.
\end{aligned}$$

Notice that in this case we can choose an arbitrary interior feasible X_0 and arbitrary (y_0, Z_0) , independent of X_0 , such that $\sum_{i=1}^m (y_0)_i A_i + Z_0 = C$. By choosing $T = (1/\mu)Z_0 - X_0^{-1}$, we obtain an affine scaling trajectory that passes through (X_0, y_0, Z_0) . We note that for a given (X_0, y_0) this matrix T is the same as the matrix T computed using (13). All results on the limiting behavior of the infeasible central path hold for these primal affine scaling trajectories. Clearly, the choice of (X_0, y_0, Z_0) affects only the limit of the primal solutions, since the dual solutions converge to the analytic center (not shifted) of the dual optimal face.

It is easy to see that system (17) holds for $(X(\mu), y(\mu), Z(\mu))$ defined by (20). Thus the tangent $\dot{X}$ to the primal trajectory $X(\mu)$ can be determined by applying the affine scaling H to the solution Δ to problem (19). Both (19) and H depend only on the primal solution $X(\mu)$. Thus, given a starting feasible point X_0 , the primal trajectories defined by (20) and (ICP_μ) coincide, since their tangents are unique solutions to the same problem. Moreover, these primal trajectories and those determined by solutions to (ICP_μ) for infeasible starting solutions X_0 depend only on the choice of X_0 and do not depend on the choice of the starting dual solution (y_0, Z_0) .

6. Derivatives of the central and infeasible central paths. We begin this section by showing that, as in case of LP [17], the computation of derivatives of any order of solutions on the central path or an infeasible central path involves inverting a single matrix. In contrast with LP, the Schur complement of this matrix, which we must factorize, is fully dense even if the constraint matrices are sparse. Consequently, this factorization step is very expensive in SDP and it is desirable to use as much information (e.g., higher order derivatives) as possible from it, as in the interior point LP methods proposed in [20] and [18].

Let us consider solutions X and (y, Z) on the infeasible central path corresponding to a shift T . (X, y , and Z depend on μ , but we omit the argument.) As shown in the previous section (see (16)), the derivatives $\dot{X}$ and $\dot{y}$ in vector form satisfy

$$(21) \quad \begin{bmatrix} \mu X^{-1} \otimes X^{-1} & -A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \text{vec}(\dot{X}) \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \text{vec}(X^{-1} + T) \\ -d \end{bmatrix}.$$

For the second derivatives we have

$$\begin{bmatrix} \mu X^{-1} \otimes X^{-1} & -A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \text{vec}(\ddot{X}) \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} 2\text{vec}(\mu X^{-1} \dot{X} X^{-1} \dot{X} X^{-1} - X^{-1} \dot{X} X^{-1}) \\ 0 \end{bmatrix},$$

and it can be easily shown by induction that the k th derivatives $X^{(k)}$ and $y^{(k)}$ on the central path must satisfy a system of equations of the form

$$\begin{bmatrix} \mu X^{-1} \otimes X^{-1} & -A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \text{vec}(X^{(k)}) \\ y^{(k)} \end{bmatrix} = \text{vec}(R),$$

where R is a function of $(T, X, \dot{X}, \ddot{X}, \dots, X^{(k-1)})$.

We now turn to the limiting properties of the first order derivatives of the central path. As earlier, let $(\bar{X}, \bar{y}, \bar{Z})$ be the limit of the central path. We need the following assumption.

ASSUMPTION 6.1.

- i. *The primal and dual solutions $\bar{X}$ and $(\bar{y}, \bar{Z})$ are strictly complementary, i.e., $r + s = n$.*
- ii. *The primal solution $\bar{X}$ is nondegenerate; i.e., the matrices*

$$B_i = \begin{bmatrix} \bar{Q}_P^T A_i \bar{Q}_P & \bar{Q}_P^T A_i \bar{Q}_D \\ \bar{Q}_D^T A_i \bar{Q}_P & 0 \end{bmatrix}, \quad i = 1, \dots, m,$$

are linearly independent in $\mathbf{S}^{n \times n}$.

- iii. *The dual solution $(\bar{y}, \bar{Z})$ is nondegenerate; i.e., the matrices*

$$B_i = [\bar{Q}_P^T A_i \bar{Q}_P], \quad i = 1, \dots, m,$$

span $\mathbf{S}^{r \times r}$.

For these and other equivalent definitions of primal and dual nondegeneracy see Alizadeh, Haeberly, and Overton [3]. They also prove that, under Assumption 6.1, the optimal primal-dual solution is unique. Hence if Assumption 6.1 holds it makes sense to say that problems (P) and (D) are nondegenerate and strictly complementary. For the remainder of this section we shall assume that Assumption 6.1 holds.

To show that the first order derivatives of the central path converge to finite limits as $\mu \rightarrow 0$, we consider the following system which, as shown in [4], is equivalent to (CP_μ) :

$$(22) \quad \begin{aligned} \frac{1}{2}(XZ + ZX) &= \mu I, \\ A_i \bullet X &= b_i, \quad i = 1, \dots, m, \\ \sum_{i=1}^m y_i A_i + Z &= C, \\ X, Z &\succ 0, \quad X, Z \in \mathbf{S}^{n \times n}. \end{aligned}$$

If we differentiate the first equation of the above system, we get

$$\frac{1}{2}(\dot{X}Z + X\dot{Z} + \dot{Z}X + Z\dot{X}) = I.$$

Therefore, viewing all symmetric matrices in (22) as vectors in $\mathbf{R}^{n(n+1)/2}$ and differentiating, we obtain the following system:

$$(23) \quad \begin{bmatrix} Z * I & 0 & X * I \\ A & 0 & 0 \\ 0 & A^T & I \end{bmatrix} \begin{bmatrix} \text{svec}(\dot{X}) \\ \dot{y} \\ \text{svec}(\dot{Z}) \end{bmatrix} = \begin{bmatrix} \text{svec}(I) \\ 0 \\ 0 \end{bmatrix},$$

where **svec** maps a matrix in $\mathbf{S}^{n \times n}$ into a vector in $\mathbf{R}^{n(n+1)/2}$ and $*$ denotes “symmetric” Kronecker product. See [4] for definitions and properties of **svec** and $*$. Alizadeh, Haeberly, and Overton in [4] prove that under Assumption 6.1 the coefficient matrix of (23) is nonsingular at the limit $\mu = 0$. Therefore, $\dot{X}$, $\dot{y}$, and $\dot{Z}$ are bounded and converge as $\mu \rightarrow 0$. Geometrically this means that the central path approaches the boundary of the feasible region at a strictly positive angle, i.e., the angle between the

tangent to the central path and the tangent to the boundary at the optimal solution is strictly positive. From $X \rightarrow \bar{X}$ and the boundedness of the derivative of X as $\mu \rightarrow 0$, we conclude that $\dot{\bar{X}} \equiv \lim_{\mu \rightarrow 0} \dot{X}(\mu) = \dot{X}(0) \equiv \dot{\bar{X}}$, where $\dot{X}(0)$ is the right derivative of $X(\mu)$ at $\mu = 0$.

Similar arguments prove that the derivatives of the primal and dual solutions on the infeasible central path are bounded and converge as $\mu \rightarrow 0$. We obtain a system similar to (23), with the same coefficient matrix and right-hand side, which depends on the shift T and on X and is uniformly bounded as $\mu \rightarrow 0$.

Let us consider the derivatives of the eigenvalues of X and Z on the central path. Since $X = X(\mu)$ is symmetric and is a smooth function of a single parameter, its eigenvalues can be ordered so that $\lambda_i = \lambda_i(\mu)$ of X is in C^1 [12]. If λ_i is an eigenvalue of multiplicity k , then the vector of derivatives $(\dot{\lambda}_{i_1}, \dots, \dot{\lambda}_{i_k})^T$ of the set of k eigenvalues $(\lambda_{i_1}, \dots, \lambda_{i_k})$ that have the value λ_i is the vector of eigenvalues of the matrix

$$(24) \quad Q_i^T \dot{X} Q_i,$$

where Q_i is a matrix of k mutually orthonormal eigenvectors of X corresponding to λ_i . If $X \rightarrow \bar{X}$ and $\dot{X} \rightarrow \dot{\bar{X}}$ as $\mu \rightarrow 0$, then Q_i does not generally converge; however, the subspace spanned by the columns of Q_i converges as $\mu \rightarrow 0$ and therefore any accumulation point of the matrices (24) has the same eigenvalues. Thus the limit of $\dot{\lambda}_i$ exists. The same can be shown for the derivatives $\dot{\omega}_i$ of the eigenvalues ω_i of the dual slack matrix Z .

Let us consider the eigenvalues of X and Z that converge to zero. The multiplicity of the zero eigenvalue of $\bar{X}$ equals s . The multiplicity of the zero eigenvalue of $\bar{Z}$ is r . We know that, for X and Z on the central path,

$$XZ = \mu I,$$

and therefore,

$$(25) \quad \dot{X}Z + X\dot{Z} = I.$$

Since $X \rightarrow \bar{X} = \bar{Q}\bar{\Lambda}\bar{Q}^T$, $Z \rightarrow \bar{Z} = \bar{Q}\bar{\Omega}\bar{Q}^T$, $\dot{X} \rightarrow \dot{\bar{X}}$, and $\dot{Z} \rightarrow \dot{\bar{Z}}$ as $\mu \rightarrow 0$,

$$\bar{Q}^T \dot{\bar{X}} \bar{Q} \bar{Q}^T \bar{Z} \bar{Q} + \bar{Q}^T \bar{X} \bar{Q} \bar{Q}^T \dot{\bar{Z}} \bar{Q} = \bar{Q}^T \dot{\bar{X}} \bar{Q} \bar{\Omega} + \bar{\Lambda} \bar{Q}^T \dot{\bar{Z}} \bar{Q} = I.$$

Considering the above system in more detail, we obtain

$$\begin{bmatrix} \bar{Q}_P^T \dot{\bar{X}} \bar{Q}_P & \bar{Q}_P^T \dot{\bar{X}} \bar{Q}_D \\ \bar{Q}_D^T \dot{\bar{X}} \bar{Q}_P & \bar{Q}_D^T \dot{\bar{X}} \bar{Q}_D \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & \bar{\Omega}_D \end{bmatrix} + \begin{bmatrix} \bar{\Lambda}_P & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{Q}_P^T \dot{\bar{Z}} \bar{Q}_P & \bar{Q}_P^T \dot{\bar{Z}} \bar{Q}_D \\ \bar{Q}_D^T \dot{\bar{Z}} \bar{Q}_P & \bar{Q}_D^T \dot{\bar{Z}} \bar{Q}_D \end{bmatrix} = I,$$

where $\bar{\Lambda}_P \succ 0$ and $\bar{\Omega}_D \succ 0$. From the diagonal blocks of this matrix equation, it then follows that

$$\bar{Q}_D^T \dot{\bar{X}} \bar{Q}_D = \bar{\Omega}_D^{-1}$$

and

$$\bar{Q}_P^T \dot{\bar{Z}} \bar{Q}_P = \bar{\Lambda}_P^{-1}.$$

Therefore, we can conclude that there are orderings of the eigenvalues of $\dot{\bar{X}}$ and $\dot{\bar{Z}}$, for which $\dot{\bar{\Lambda}}_D \rightarrow \bar{\Omega}_D^{-1}$ and $\dot{\bar{\Omega}}_P \rightarrow \bar{\Lambda}_P^{-1}$. This generalizes the result on the limits of the

derivatives of nonbasic variables (i.e., variables converging to zero) in LP. Complete information on $\dot{\bar{X}}$ and $\dot{\bar{Z}}$ can be obtained by solving the system (23).

We can now conclude that the dual estimates $y^E = y - \mu \dot{y}$ and $Z^E = Z - \mu \dot{Z}$ that appear in (18) converge to the same limits as y and Z , since $\dot{y}$ and $\dot{Z}$ are bounded as $\mu \rightarrow 0$.

Remark. To prove the convergence of the derivatives of the central path as $\mu \rightarrow 0$, we assumed primal and dual nondegeneracy and strict complementarity. These assumptions imply the uniqueness of the primal and dual solutions [3]. However, we conjecture that it is sufficient to only assume strict complementarity. In [16] it is shown that, assuming strict complementarity, $\|X(\mu) - \bar{X}\| = O(\mu)$ and $\|Z(\mu) - \bar{Z}\| = O(\mu)$, which imply that the derivatives of the central path are bounded as $\mu \rightarrow 0$. Note that strict complementarity is necessary for boundedness of the derivatives. Indeed, suppose it does not hold; then $\mathbf{R}_N \neq \{0\}$. From (25) we have

$$Q_N^T(\mu) \dot{X}(\mu) Q_N(\mu) Q_N^T(\mu) Z(\mu) Q_N(\mu) + Q_N^T(\mu) X(\mu) Q(\mu) Q_N^T(\mu) \dot{Z}(\mu) Q_N(\mu) = I.$$

Since $Q_N^T(\mu) Z(\mu) Q_N(\mu) \rightarrow 0$ and $Q_N^T(\mu) X(\mu) Q(\mu) \rightarrow 0$ as $\mu \rightarrow 0$, $Q_N^T(\mu) \dot{X}(\mu) Q_N(\mu)$ and $Q_N^T(\mu) \dot{Z}(\mu) Q_N(\mu)$ cannot be finite in the limit as $\mu \rightarrow 0$.

Acknowledgments. The authors are grateful to Yurii Nesterov for bringing the question of the convergence of the central path in SDP to their attention, to Michael Overton for many stimulating discussions on SDP, and to Jean-Pierre Haeberly, Maza-kazu Kojima, and two anonymous referees for helpful comments on earlier versions of the paper.

REFERENCES

- [1] I. ADLER AND R. D. C. MONTEIRO, *Limiting behavior of the affine scaling continuous trajectories for linear programming problems*, Math. Programming, 50 (1991), pp. 29–51.
- [2] F. ALIZADEH, *Interior point methods in semidefinite programming with applications to combinatorial optimization*, SIAM J. Optim., 5 (1995), pp. 13–51.
- [3] F. ALIZADEH, J.-P. A. HAEBERLY, AND M. L. OVERTON, *Complementarity and nondegeneracy in semidefinite programming*, Math. Programming, 77 (1997), pp. 111–128.
- [4] F. ALIZADEH, J.-P. A. HAEBERLY, AND M. L. OVERTON, *Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results*, SIAM J. Optim., 8 (1998), pp. 746–768.
- [5] G. PH. BARKER AND D. CARLSON, *Cones of diagonally dominant matrices*, Pacific J. Math., 57 (1975), pp. 15–32.
- [6] D. A. BAYER AND J. C. LAGARIAS, *The nonlinear geometry of linear programming. I. Affine and projective scaling trajectories*, Trans. Amer. Math. Soc., 314 (1989), pp. 499–526.
- [7] L. FAYBUSOVICH, *On a matrix generalization of affine-scaling vector fields*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 886–897.
- [8] L. FAYBUSOVICH, *A Hamiltonian structure of generalized affine scaling vector fields*, J. Nonlinear Sci., 5 (1995), pp. 11–28.
- [9] A. GRAHAM, *Kronecker Products and Matrix Calculus: With Applications*, Ellis Horwood, London, UK, 1981.
- [10] O. GÜLER, *Existence of interior points and interior paths in nonlinear monotone complementarity problems*, Math. Oper. Res., 18 (1993), pp. 128–147.
- [11] C. HELMBERG, F. RENDL, R. VANDERBEI, AND H. WOLKOWICZ, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.
- [12] T. KATO, *A Short Introduction to Perturbation Theory for Linear Operators*, Springer-Verlag, New York, 1982.
- [13] M. KOJIMA, S. MIZUNO, AND T. NOMA, *Limiting behavior of trajectories generated by a continuation method for monotone complementarity problems*, Math. Oper. Res., 15 (1990), pp. 662–675.

- [14] M. KOJIMA, M. SHIDA, AND S. SHINDOH, *Local convergence of predictor-corrector infeasible-interior-point algorithms for SDPs and SDLCPs*, Math. Programming, 80 (1998), pp. 129–160.
- [15] M. KOJIMA, S. SHINDOH, AND S. HARA, *Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.
- [16] Z.-Q. LUO, J. STURM, AND S. ZHANG, *Superlinear Convergence of a Symmetric Primal-dual Path Following Algorithm for Semidefinite Programming*, manuscript, Department of Electrical and Computer Engineering, McMaster University, Ontario, Canada and Econometric Institute, Erasmus University, Rotterdam, The Netherlands, 1996.
- [17] N. MEGIDDO, *Pathways to the optimal set in linear programming*, in Progress in Mathematical Programming: Interior-Point Algorithms and Related Methods, N. Megiddo, ed., Springer-Verlag, Berlin, 1989, pp. 131–158.
- [18] S. MEHROTRA, *Higher Order Methods and Their Performance*, Technical Report 90-16R1, Dept. of IE and MS, Northwestern Univ., Evanston, IL, 1991.
- [19] R. D. C. MONTEIRO, *Primal-dual path following algorithms for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 663–678.
- [20] R. D. C. MONTEIRO, I. ADLER, AND M. G. C. RESENDE, *A polynomial time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extension*, Math. Oper. Res., 15 (1990), pp. 191–214.
- [21] YU. E. NESTEROV AND A. S. NEMIROVSKII, *Interior point Methods in Convex Programming: Theory and Application*, SIAM, Philadelphia, PA, 1994.
- [22] YU. E. NESTEROV AND M. J. TODD, *Self-scaled cones and interior-point methods in nonlinear programming*, Math. Oper. Res., 22 (1997), pp. 1–42.
- [23] YU. E. NESTEROV AND M. J. TODD, *Primal-dual interior point algorithms for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.
- [24] G. PATAKI, *On the Facial Structure of Cone-LP's and Semi-definite Programs*, Research Report MSRR-595, Graduate School of Industrial Administration, Carnegie Mellon Univ., Pittsburgh, PA, 1994.
- [25] F. A. POTRA AND R. SHENG, *Superlinear Convergence of Interior-point Algorithms for Semidefinite Programming*, Research Report 86, Dept. of Mathematics, University of Iowa, IA, 1996.
- [26] M. RAMANA, L. TUNÇEL, AND H. WOLKOWICZ, *Strong duality for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 641–662.
- [27] K. SCHEINBERG, *Issues in Interior Point Methods in Semidefinite and Linear Programming*, Ph.D. thesis, Columbia University, New York, 1996.
- [28] M. SHIDA AND S. SHINDOH, *Monotone Semidefinite Complementarity Problems*, Working Paper B-312, Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, Japan, 1996.
- [29] L. VANDERBERGHE AND S. BOYD, *A primal-dual potential reduction method for problems involving matrix inequalities*, Math. Programming, 69 (1995), pp. 205–236.
- [30] L. VANDERBERGHE AND S. BOYD, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.
- [31] Y. YE, *Convergence Behavior on Central Paths for Convex Homogeneous Self-dual Cones*, note, Dept. of Managements, University of Iowa, IA, 1995.
- [32] Y. ZHANG, *On extending some primal-dual interior-point algorithms from linear programming to semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 365–386.

ON THE CONVERSION OF OPTIMIZATION PROBLEMS WITH MAX-MIN CONSTRAINTS TO STANDARD OPTIMIZATION PROBLEMS*

C. KIRJNER-NETO[†] AND E. POLAK[†]

Abstract. We consider the problem of minimizing a real-valued function $f^0(x)$ subject to max-min constraints, i.e., constraints of the form $\max_{k \in \mathbf{r}} \min_{j \in \mathbf{q}_k} f_k^j(x) \leq 0$, where for any integer r , $\mathbf{r} \triangleq \{1, \dots, r\}$, so that for any $k \in \mathbf{r}$, $\mathbf{q}_k = \{1, \dots, q_k\}$. Problems with this kind of constraint arise in a variety of applications, such as design of electronic circuits subject to manufacturing tolerances and postmanufacturing tuning and optimal steering of mobile robots in the presence of obstacles.

An optimization problem with max-min constraints is equivalent to a collection of optimization problems with smooth inequality constraints. This collection can be very large, and hence solving an optimization problem with max-min constraints by solving each member of the collection can be exorbitantly expensive. We present a transcription of problems of this type into a single inequality constrained nonlinear programming problem with smooth constraints, which can be solved using readily available software. The transcription is based on the fact that a set of real numbers contains a nonpositive element if and only if its convex hull contains a nonpositive element. While this fact is fairly obvious, establishing a correspondence between the global and local minimizers of the original problem and those of the problem resulting from our transcription turned out to be quite difficult.

The significant advantages of using our transcription are demonstrated by the numerical examples that are included in the paper to illustrate the effectiveness of our approach.

Key words. max-min constraints, disjunctive optimization, nonlinear programming, path-planning

AMS subject classifications. Primary, 65K05; Secondary, 90C30

PII. S1052623496298534

1. Introduction. Optimization problems with max-min constraints are well known in engineering, since they arise in the design of electronic circuits subject to a tolerancing and tuning provision [BaL.1, LiC.1, Mul.1, PoS.1, Pol.1], in the design of paths for robots in the presence of obstacles [GiJ.1], in the design of heat exchangers [GrS.1, HaG.1, OsV.1] and chemical reactors [HaG.1, OsV.1], in the layout design of VLSI circuits [ChD.1, Hoc.1], etc. (see [IeP.1, BrM.1] for problems with max-min constraints in other areas). These problems have constraints of the form $\max_{k \in \mathbf{r}} \min_{j \in \mathbf{q}_k} f_k^j(x) \leq 0$, where for any integer r , $\mathbf{r} \triangleq \{1, \dots, r\}$, so that for any $k \in \mathbf{r}$, $\mathbf{q}_k = \{1, \dots, q_k\}$. Such problems are virtually unknown to the mathematical programming community. A sense of how such problems arise can be obtained by considering the following simple design problem, subject to tolerancing and tuning. One is required to compute a “design center” $x \in \mathbb{R}^n$ that minimizes a continuously differentiable objective function $f^0: \mathbb{R}^n \rightarrow \mathbb{R}$ while guaranteeing that a design specification is satisfied. Physically, this center can only be manufactured to a tolerance ε , and after manufacture some of the effects of the tolerances can be mitigated by “tuning.” The range of tuning is governed by the admissible values of a tuning parameter τ . Since the effect of tolerances makes satisfaction of design constraints more difficult,

*Received by the editors February 9, 1996; accepted for publication (in revised form) June 10, 1997; published electronically August 26, 1998. This research was supported by NSF grant ECS-9302926 and CUREe-Kajima grant M1975.

<http://www.siam.org/journals/siopt/8-4/29853.html>

[†]Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720 (kirjner@eecs.berkeley.edu, polak@optimum.eecs.berkeley.edu).

while tuning has the opposite effect, we get an optimization problem of the form

$$(1.1) \quad \min_x \{f^0(x) \mid \max_{\varepsilon \in E} \min_{\tau \in T} f(x + \varepsilon + \tau) \leq 0\},$$

where the continuously differentiable function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ is a design constraint function (it takes a nonpositive value if and only if the design corresponding to its argument satisfies the specification), E is the range to which the tolerancing parameter is confined, and T is the range to which the tuning parameter is confined. If we discretize the sets E and T , we eventually get a problem of the form

$$(1.2) \quad (\text{P}) \quad \min_x \{f^0(x) \mid \max_{k \in \mathbf{r}} \min_{j \in \mathbf{q}_k} f_k^j(x) \leq 0\},$$

where the functions $f_k^j: \mathbb{R}^n \rightarrow \mathbb{R}$, $k \in \mathbf{r}$, $j \in \mathbf{q}_k$, are also continuously differentiable.

Problem (P) corresponds to a collection of optimization problems with smooth inequality constraints. Indeed, let us consider the case where $r = q_1 = q_2 = 2$, i.e., the case where (P) may be written as

$$(1.3a) \quad \min \{f^0(x) \mid \min \{f_1^1(x), f_1^2(x)\} \leq 0 \text{ and } \min \{f_2^1(x), f_2^2(x)\} \leq 0\}.$$

One can decompose this optimization problem into the following collection of subproblems:

$$(1.3b) \quad \min \{f^0(x) \mid f_1^1(x) \leq 0, f_1^2(x) \leq 0\}, \quad \min \{f^0(x) \mid f_1^1(x) \leq 0, f_2^2(x) \leq 0\},$$

$$(1.3c) \quad \min \{f^0(x) \mid f_1^2(x) \leq 0, f_1^1(x) \leq 0\}, \quad \min \{f^0(x) \mid f_1^2(x) \leq 0, f_2^2(x) \leq 0\}.$$

One strategy for solving (1.3a) is to solve each of the subproblems in (1.3b–c) and choose, among their solutions, the minimizer $\hat{x}$ that corresponds to the least value of the objective function $f^0(\cdot)$. There are two major objections to this decomposition strategy.

The first is that some of the subproblems may have an empty feasible set, even when the feasible set for the original problem with max-min constraints is not empty (see section 4.1 for an example). This has an undesirable practical consequence: many algorithms may require a relatively large number of function and gradient evaluations before stopping because a feasible point cannot be computed. Moreover, it is often impossible to ascertain whether an algorithm fails to compute a feasible point because the feasible set is empty, because a problem is very difficult to solve (e.g., poorly conditioned), or because the initial guess is bad.

The second and perhaps most important objection to the above decomposition strategy stems from the fact that the number of subproblems to be solved grows very rapidly as the number of min constraints, r , increases. If $q_k = q > 1$ for all $k \in \mathbf{r}$, then the number of subproblems to be solved is q^r , which grows exponentially with r . Therefore, the decomposition strategy becomes exorbitantly expensive even for relatively small values of r : if $r = 20$ and $q = 2$, there are 2^{20} , i.e., more than a million, subproblems to be solved. Problems with r considerably larger than 20 are easily generated by discretizing optimal control and semi-infinite optimization problems, as we will see in section 4.

The literature dealing with problems of the form (P) is relatively small. To the authors' knowledge, the only implementable algorithms to solve (P) without assuming linearity or convexity of its defining functions are those in [Tit.1] and [MaP.1]. A practical disadvantage of these algorithms is that they cannot be implemented by

straightforward utilization of existing nonlinear programming code. Indeed, to the best of our knowledge the algorithms proposed in [Tit.1, MaP.1] have never been implemented, and hence their numerical advantages have not been established.

In this paper we present a simple transcription of (P) into a single nonlinear programming problem ($\tilde{P}$) with smooth constraints, which can be solved using a number of existing software packages for nonlinear programming. The transcription is based on the fact that a set of real numbers contains a nonpositive element if and only if its convex hull contains a nonpositive element. While this fact is fairly obvious, establishing a correspondence between (P) and ($\tilde{P}$) turned out to be quite difficult. We prove equivalence, in a sense to be explained below, between global and local minimizers of (P) and those of ($\tilde{P}$). Global and local minimizers of (P) can be obtained in a straightforward fashion from the corresponding global and local minimizer of ($\tilde{P}$), and hence one can compute a solution for (P) by solving ($\tilde{P}$) instead.

The paper is organized as follows. In section 2, we reformulate the max-min constraints of (P) to obtain a new problem ($\tilde{P}$) with smooth inequality constraints. We then establish equivalence between the global minimizers of (P) and those of ($\tilde{P}$). In section 3, we show equivalence between local minimizers of (P) and ($\tilde{P}$). The results of numerical experiments presented in section 4 illustrate the efficiency of solving (P) by means of the proposed reformulation. Section 5 contains a few concluding remarks.

2. Formulation of the equivalent problem $\tilde{P}$. What sets (P) apart from the “standard” nonlinear programming problem with smooth inequality constraints is the “min” operator in the definition of its constraints. Hence, to reformulate P into a more tractable form, for any positive integer r , let $\mathbf{r} = \{1, \dots, r\}$, and, with r given, let $k \in \mathbf{r}$ be fixed and consider the constraint

$$(2.1) \quad \min_{j \in \mathbf{q}_k} f_k^j(x) \leq 0.$$

Given an integer q , let Σ_q denote the q -simplex, i.e.,

$$(2.2) \quad \Sigma_q \triangleq \left\{ \mu \in \mathbb{R}^q \left| \sum_{j \in \mathbf{q}} \mu^j = 1, \mu^j \geq 0, j \in \mathbf{q} \right. \right\}.$$

On the one hand, we have that a collection $\{y_i\}_{i=1}^{q_k}$ of real numbers contains a nonpositive number if and only if there exists a $\mu \in \Sigma_{q_k}$ such that

$$(2.3) \quad \sum_{i \in \mathbf{q}_k} \mu^i y_i \leq 0.$$

On the other hand, $x \in \mathbb{R}^n$ satisfies (2.1) if and only if one of the real numbers $\{f_k^j(x)\}_{j=1}^{q_k}$ is nonpositive. Hence, $x \in \mathbb{R}^n$ satisfies (2.1) if and only if there exists $\mu_k \in \Sigma_{q_k}$ such that

$$(2.4) \quad \sum_{j \in \mathbf{q}_k} \mu_k^j f_k^j(x) \leq 0.$$

Therefore, if we define

$$(2.5) \quad (\tilde{P}) \quad \min_{(x, \mu_1, \dots, \mu_r)} \left\{ f^0(x) \left| \sum_{j \in \mathbf{q}_k} \mu_k^j f_k^j(x) \leq 0, \mu_k \in \Sigma_{q_k}, k \in \mathbf{r} \right. \right\},$$

we immediately have the following result.

PROPOSITION 2.1. $x \in \mathbb{R}^n$ is feasible for (P) if and only if there exists $\mu_k \in \Sigma_{q_k}$, $k \in \mathbf{r}$, such that $(x, \mu_1, \dots, \mu_r)$ is feasible for $(\tilde{P})$.

As a consequence of Proposition 2.1, we have that the global minimizers of P and $\tilde{P}$ are equivalent in the following sense.

THEOREM 2.2. $\hat{x} \in \mathbb{R}^n$ is a global minimizer for (P) if and only if there exists $\mu_k \in \Sigma_{q_k}$, $k \in \mathbf{r}$, such that $(\hat{x}, \hat{\mu}_1, \hat{\mu}_2, \dots, \hat{\mu}_r)$ is a global minimizer for $(\tilde{P})$.

Proof. The result follows from Proposition 2.1 and from the fact that the objective function of $(\tilde{P})$ does not depend on the μ 's. Indeed, assume that $\hat{x}$ is a global minimizer for $(\tilde{P})$. Then $\hat{x}$ is feasible for (P), and by Proposition 2.1 there exist $\hat{\mu}_k$, $k \in \mathbf{r}$, such that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is feasible for $(\tilde{P})$. To obtain a contradiction, suppose that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is not a global minimizer for $(\tilde{P})$. Then there exists $(x^*, \mu_1^*, \dots, \mu_r^*)$, feasible for $(\tilde{P})$ and such that $f^0(x^*) < f^0(\hat{x})$. It follows from Proposition 2.1 that x^* is feasible for (P), and $f^0(x^*) < f^0(\hat{x})$. Hence, $\hat{x}$ is not a global minimizer for (P), which is a contradiction.

To prove the converse we use contraposition. Suppose that $\hat{x}$ is not a global minimizer for (P), i.e., that there exists x^* , feasible for (P), such that $f^0(x^*) < f^0(\hat{x})$. It follows from Proposition 2.1 that there exist μ_k^* , $k \in \mathbf{r}$, such that $(x^*, \mu_1^*, \dots, \mu_r^*)$ is feasible for $(\tilde{P})$, and $f^0(x^*) < f^0(\hat{x})$. Since the objective function of $(\tilde{P})$ does not depend on the μ_k 's, this clearly implies that there exists no $(\hat{\mu}_1, \dots, \hat{\mu}_r)$ such that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a global minimizer for $(\tilde{P})$, which in turn implies that the desired result holds. $\square$

Thus, the conversion of problem (P) into the standard nonlinear programming problem $(\tilde{P})$ was achieved at the cost of $\sum_{k=1}^r q_k$ extra variables and r additional constraints.

3. Correspondence of local minimizers. We will show in this section that the local minimizers of (P) and $(\tilde{P})$ are equivalent. More precisely, we will establish that $\hat{x}$ is a local minimizer of (P) if and only if there exist $\hat{\mu}_k \in \Sigma_{q_k}$, $k \in \mathbf{r}$, such that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer of $(\tilde{P})$. As we will soon see, it is rather easy to show that every local minimizer of (P) corresponds to a local minimizer of $(\tilde{P})$. However, to show the converse is more difficult. We start by introducing some notation.

DEFINITION 3.1. (a) For each $k \in \mathbf{r}$, let

$$(3.1a) \quad \psi_k(x) \triangleq \min_{j \in \mathbf{q}_k} f_k^j(x),$$

$$(3.1b) \quad \hat{\mathbf{q}}_k(x) \triangleq \{j \in \mathbf{q}_k \mid f_k^j(x) = \psi_k(x)\},$$

$$(3.1c) \quad \hat{\mathbf{q}}_k^c(x) \triangleq \mathbf{q}_k \setminus \hat{\mathbf{q}}_k(x).$$

The cardinality of $\hat{\mathbf{q}}_k(x)$ will be denoted by $|\hat{\mathbf{q}}_k(x)|$.

(b) We define

$$(3.1d) \quad \psi(x) \triangleq \max_{k \in \mathbf{r}} \psi_k(x),$$

$$(3.1e) \quad \hat{\mathbf{r}}(x) \triangleq \{k \in \mathbf{r} \mid \psi_k(x) = \psi(x)\},$$

$$(3.1f) \quad \hat{\mathbf{r}}^c(x) \triangleq \mathbf{r} \setminus \hat{\mathbf{r}}(x).$$

The cardinality of $\hat{\mathbf{r}}(x)$ is denoted by $|\hat{\mathbf{r}}(x)|$.

THEOREM 3.2. *If $\hat{x} \in \mathbb{R}^n$ is a local minimizer for (P), then there exist $\mu_k \in \Sigma_{q_k}$, $k \in \mathbf{r}$, such that $(\hat{x}, \hat{\mu}_1, \hat{\mu}_2, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{P})$.*

Proof. Assume that $\hat{x}$ is a local minimizer for (P). For each $k \in \mathbf{r}$, let $\hat{\mu}_k \in \Sigma_{q_k}$ be such that $\sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \hat{\mu}_k^j = 1$. Since $\hat{x}$ is a local minimizer for (P), it must be feasible for (P). Hence, we have that $f_k^j(\hat{x}) = \psi_k(\hat{x}) \leq 0$ for each $k \in \mathbf{r}$ and each $j \in \hat{\mathbf{q}}_k(\hat{x})$. It follows from the choice of $(\hat{\mu}_1, \dots, \hat{\mu}_r)$ that, for each $k \in \mathbf{r}$,

$$(3.2) \quad \sum_{j \in \mathbf{q}_k} \hat{\mu}_k^j f_k^j(\hat{x}) = \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \hat{\mu}_k^j f_k^j(\hat{x}) = \psi_k(\hat{x}) \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \hat{\mu}_k^j = \psi_k(\hat{x}) \leq 0,$$

i.e., that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is feasible for $(\tilde{P})$. To complete the proof, we show that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{P})$. To obtain a contradiction, suppose that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is not a local minimizer for $(\tilde{P})$. Then there exists a sequence $\{(x_i, \mu_{1,i}, \dots, \mu_{r,i})\}_{i=0}^\infty$ converging to $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ as $i \rightarrow \infty$, feasible for $(\tilde{P})$, and such that $f^0(x_i) < f^0(\hat{x})$ for all i . Since $\{(x_i, \mu_{1,i}, \dots, \mu_{r,i})\}_{i=0}^\infty$ is feasible for $(\tilde{P})$, we have, by Proposition 2.1, that $\{x_i\}_{i=0}^\infty$ is a feasible sequence for (P) such that $f^0(x_i) < f^0(\hat{x})$ for all i , and $x_i \rightarrow \hat{x}$ as $i \rightarrow \infty$. This implies that $\hat{x}$ is not a local minimizer for (P), which is a contradiction. $\square$

Note that Theorems 2.2 and 3.2 hold without any assumptions on the smoothness properties of $f^0(\cdot)$ and $f_k^j(\cdot)$, $k \in \mathbf{r}$, $j \in \mathbf{q}_k$. We will need the following assumption to show the converse of Theorem 3.2.

ASSUMPTION 3.3. *The functions $f^0: \mathbb{R}^n \rightarrow \mathbb{R}$ and $f_k^j: \mathbb{R}^n \rightarrow \mathbb{R}$, $k \in \mathbf{r}$, $j \in \mathbf{q}_k$, are twice continuously differentiable.*

We now show that if $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{P})$, then $\hat{x}$ is a local minimizer for (P). The proof of this fact is divided into several steps. First we deal with the simpler case where $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{P})$, and $\psi(\hat{x}) < 0$; i.e., $\hat{x}$ is in the interior of the feasible set of (P). We start by establishing the following technical result.

LEMMA 3.4. *Suppose that Assumption 3.3 is satisfied, that $x \in \mathbb{R}^n$ and $k \in \mathbf{r}$ are such that $\psi_k(x) = \min_{j \in \mathbf{q}_k} f_k^j(x) < 0$, and that $\mu_k \in \Sigma_{q_k}$ is such that $\sum_{j \in \mathbf{q}_k} \mu_k^j f_k^j(x) \leq 0$. Let $\{x_i\}_{i=0}^\infty$ be a sequence in $\mathbb{R}^n$ such that $x_i \rightarrow x$ as $i \rightarrow \infty$. Then there exists an $i_0 \in \mathbb{N}$ and a sequence $\{\mu_{k,i}\}_{i=0}^\infty$ in Σ_{q_k} such that $\mu_{k,i} \rightarrow \mu_k$ as $i \rightarrow \infty$ and, for all $i \geq i_0$,*

$$(3.3a) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j f_k^j(x_i) < 0.$$

Proof. Because the functions $f_k^j(\cdot)$ are assumed to be continuous, the lemma is clearly true if $\sum_{j \in \mathbf{q}_k} \mu_k^j f_k^j(x) < 0$, since in this case we can set $\mu_{k,i} = \mu_k$ for all $i \in \mathbb{N}$. Hence, we only need to consider the case where $\sum_{j \in \mathbf{q}_k} \mu_k^j f_k^j(x) = 0$, i.e., $\langle \mu_k, F_k(x) \rangle = 0$, where

$$(3.3b) \quad F_k(x) \triangleq \begin{bmatrix} f_k^1(x) \\ \vdots \\ f_k^{q_k}(x) \end{bmatrix}.$$

Let $\varepsilon > 0$ and $j_1 \in \hat{\mathbf{q}}_k(\hat{x})$ be such that $f_k^{j_1}(x) = \min_{j \in \mathbf{q}_k} f_k^j(x) = -\varepsilon < 0$. For any $j \in \mathbf{q}_k$, let e_j denote the j th canonical basis vector in $\mathbb{R}^{q_k}$. Since $e_j \in \Sigma_{q_k}$, we have that, for any $\alpha \in (0, 1]$,

$$(3.3c) \quad (1 - \alpha)\mu_k + \alpha e_{j_1} \in \Sigma_{q_k},$$

and, since $\langle \mu_k, F_k(x) \rangle = 0$,

$$\begin{aligned} \langle (1-\alpha)\mu_k + \alpha e_{j_1}, F_k(x) \rangle &= (1-\alpha)\langle \mu_k, F_k(x) \rangle + \alpha \langle e_{j_1}, F_k(x) \rangle \\ (3.3d) \quad &= \alpha f_k^{j_1}(x) = -\alpha\varepsilon < 0. \end{aligned}$$

By continuity of the functions $f_k^j(\cdot)$, there exists an i_0 such that, for any $\alpha \in (0, 1]$ and all $i \geq i_0$,

$$(3.3e) \quad \langle (1-\alpha)\mu_k + \alpha e_{j_1}, F_k(x_i) \rangle \leq -\frac{1}{2}\alpha\varepsilon < 0.$$

Hence, setting $\alpha = 1/i$ we get

$$(3.3f) \quad \langle (1-1/i)\mu_k + e_{j_1}/i, F_k(x_i) \rangle \leq -\frac{1}{2}\varepsilon/i < 0$$

for all i . Finally, we let $\mu_{k,i} \triangleq (1-1/i)\mu_k + e_{j_1}/i$ for all $i \geq i_0$. Clearly, $\mu_{k,i} \rightarrow \mu_k$ as $i \rightarrow \infty$. Moreover, from (3.3c) and (3.3f) we have that $\mu_{k,i} \in \Sigma_{q_k}$ and

$$(3.3g) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j f_k^j(x_i) = \langle \mu_{k,i}, F_k(x_i) \rangle \leq -\frac{1}{2}\varepsilon/i < 0$$

for all $i \geq i_0$, which implies the desired result. $\square$

PROPOSITION 3.5. *Suppose that Assumption 3.3 is satisfied. If $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}})$ and $\psi(\hat{x}) < 0$, then $\hat{x}$ is a local minimizer for $(\mathbf{P})$.*

Proof. Assume that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}})$ and $\psi(\hat{x}) < 0$. First, since $\psi(\hat{x}) < 0$, it follows that, for each $k \in \mathbf{r}$,

$$(3.4a) \quad \min_{j \in \mathbf{q}_k} f_k^j(x) < 0.$$

Moreover, because $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}})$, it must be feasible for $(\tilde{\mathbf{P}})$, and hence

$$(3.4b) \quad \sum_{j \in \mathbf{q}_k} \hat{\mu}_k^j f_k^j(\hat{x}) \leq 0 \quad \text{for all } k \in \mathbf{r}.$$

Next, to obtain a contradiction suppose that $\hat{x}$ is not a local minimizer for $(\mathbf{P})$. Then there exists a sequence $\{x_i\}_{i=0}^\infty$ such that $x_i \rightarrow \hat{x}$ as $i \rightarrow \infty$, $\psi(x_i) \leq 0$ for all $i \in \mathbb{N}$, and

$$(3.4c) \quad f^0(x_i) < f^0(\hat{x})$$

for all i . Furthermore, because (3.4a) and (3.4b) hold, Lemma 3.4 implies that there exist an i_0 and sequences $\{\mu_{k,i}\}_{i=i_0}^\infty \subset \Sigma_{q_k}$, $k \in \mathbf{r}$, such that $\mu_{k,i} \rightarrow \hat{\mu}_k$ as $i \rightarrow \infty$ and

$$(3.4d) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j f_k^j(x_i) < 0 \quad \text{for all } k \in \mathbf{r}$$

and all $i \geq i_0$. Together with (3.4c), this implies that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is not a local minimizer for $(\tilde{\mathbf{P}})$, which is a contradiction. $\square$

In view of Proposition 3.5, to establish that if $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}})$, then $\hat{x}$ is a local minimizer of $(\mathbf{P})$, we need only consider the case where

$\psi(\hat{x}) = 0$. The proof for this case will proceed as follows. We will first show (see Theorem 3.8 below) that if $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{P})$ and

$$(3.5) \quad |\hat{\mathbf{q}}_k(\hat{x})| = 1 \quad \text{for all } k \in \hat{\mathbf{r}}(\hat{x}),$$

then $\hat{x}$ is a local minimizer of (P). We will then introduce a constraint qualification (stated in Assumption 3.9) and show (in Theorem 3.11) that (3.5) holds whenever this constraint qualification is satisfied. These two facts, together with Proposition 3.5, imply the converse of Theorem 3.2. The proof of Theorem 3.11 depends on a series of results which are collected in Proposition 3.10 and proved in the appendix.

Lemma 3.6 below follows directly from the definitions of $\psi(x)$, $\hat{\mathbf{r}}(x)$, $\hat{\mathbf{r}}^c(x)$, $\hat{\mathbf{q}}_k(x)$, and $\hat{\mathbf{q}}_k^c(x)$.

LEMMA 3.6. *Let $x \in \mathbb{R}^n$ be such that $\psi(x) = 0$. Then*

- (a) $\min_{j \in \mathbf{q}_k} f_k^j(x) < 0$ for all $k \in \hat{\mathbf{r}}^c(x)$.
- (b) For each $k \in \hat{\mathbf{r}}(x)$ and each $\mu_k \in \Sigma_{q_k}$, $\sum_{j \in \mathbf{q}_k} \mu_k^j f_k^j(x) \geq 0$.
- (c) $\min_{j \in \mathbf{q}_k} f_k^j(x) = 0$ for all $k \in \hat{\mathbf{r}}(x)$, and $f_k^j(x) > 0$ for all $j \in \hat{\mathbf{q}}_k^c(x)$, $k \in \hat{\mathbf{r}}(x)$.
- (d) Let $k \in \hat{\mathbf{r}}(x)$ and $\mu_k \in \Sigma_{q_k}$ be such that $\sum_{j \in \mathbf{q}_k} \mu_k^j f_k^j(x) \leq 0$. Then $\mu_k^j = 0$ for all $j \in \hat{\mathbf{q}}_k^c(x)$.

As a direct consequence of Lemma 3.6(d) we have the following result.

COROLLARY 3.7. *Suppose that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is feasible for $(\tilde{P})$ and that $\psi(\hat{x}) = 0$. Then $\hat{\mu}_k^j = 0$ for all $j \in \hat{\mathbf{q}}_k^c(\hat{x})$, $k \in \hat{\mathbf{r}}(\hat{x})$.*

THEOREM 3.8. *Suppose that Assumption 3.3 is satisfied, that $\psi(\hat{x}) = 0$, and that $|\hat{\mathbf{q}}_k(\hat{x})| = 1$ for all $k \in \hat{\mathbf{r}}(\hat{x})$. If $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{P})$, then $\hat{x}$ is a local minimizer for (P).*

Proof. In view of the correspondence between the feasible sets of (P) and $(\tilde{P})$, established in Proposition 2.1, it is clear that we need only consider the case where $\hat{x}$ is feasible for (P).

We prove the result by contraposition. Hence, suppose that $\hat{x}$ is feasible for (P) but that it is not a local minimizer for (P). We show that there do not exist $(\hat{\mu}_1, \dots, \hat{\mu}_r)$ such that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{P})$. Indeed, suppose that $(\hat{\mu}_1, \dots, \hat{\mu}_r)$ is such that

$$(3.6a) \quad \sum_{j \in \mathbf{q}_k} \hat{\mu}_k^j f_k^j(\hat{x}) \leq 0$$

for all $k \in \mathbf{r}$ and all $j \in \mathbf{q}_k$.

We prove that there exist an $i_0 \in \mathbb{N}$ and an infinite sequence $\{(x_i, \mu_{1,i}, \dots, \mu_{r,i})\}_{i=i_0}^\infty$, with $\{\mu_{k,i}\}_{i=i_0}^\infty \subset \Sigma_{q_k}$, $k \in \mathbf{r}$, converging to $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ as $i \rightarrow \infty$, such that

$$(3.6b) \quad f^0(x_i) < f^0(\hat{x})$$

and

$$(3.6c) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j f_k^j(x_i) \leq 0, \quad k \in \mathbf{r},$$

for all $i \geq i_0$.

Indeed, since $\hat{x}$ is not a local minimizer for (P) by assumption, there exists a sequence $\{x_i\}_{i=0}^\infty$, feasible for (P), converging to $\hat{x}$ as $i \rightarrow \infty$, and such that (3.6b) holds for all i .

To construct the sequences $\{\mu_{k,i}\}$, $k \in \mathbf{r}$, so that (3.6c) is satisfied, we consider separately the two possibilities $k \in \hat{\mathbf{r}}^c(\hat{x})$ and $k \in \hat{\mathbf{r}}(\hat{x})$.

First, suppose that $k \in \hat{\mathbf{r}}^c(\hat{x})$. Because $\psi(\hat{x}) = 0$, Lemma 3.6(a) implies that

$$(3.6d) \quad \psi_k(\hat{x}) = \min_{j \in \mathbf{q}_k} f_k^j(\hat{x}) < 0, \quad k \in \hat{\mathbf{r}}^c(\hat{x}).$$

Therefore, by (3.6a) and Lemma 3.4, there exists an i_0 and sequences $\{\mu_{k,i}\}_{i=i_0}^\infty \subset \Sigma_{q_k}$, $k \in \hat{\mathbf{r}}^c(\hat{x})$, such that $\mu_{k,i} \rightarrow \hat{\mu}_k$ as $i \rightarrow \infty$, and (3.6c) holds for all $k \in \hat{\mathbf{r}}^c(\hat{x})$ and all $i \geq i_0$.

Second, suppose that $k \in \hat{\mathbf{r}}(\hat{x})$. Recall that $|\hat{\mathbf{q}}_k(\hat{x})| = 1$ by assumption. Let $j_k \in \mathbf{q}_k$ be such that $\hat{\mathbf{q}}_k(\hat{x}) = \{j_k\}$. It follows from Corollary 3.7 that we must have $\hat{\mu}_k^{j_k} = 0$ for all $j \neq j_k$, and $\hat{\mu}_k^{j_k} = 1$. Moreover, because $\{x_i\}_{i=0}^\infty$ is feasible for (P), Proposition 2.1 implies that there exists a sequence $\{\mu_{k,i}\}_{i=0}^\infty \subset \Sigma_{q_k}$ satisfying (3.6c) for all i . Because Σ_{q_k} is compact we can assume, without loss of generality (one can always pass to a subsequence if necessary), that $\mu_{k,i} \rightarrow \bar{\mu}_k \in \Sigma_{q_k}$ as $i \rightarrow \infty$. It follows by taking limits on both sides of (3.6c) that we have have

$$(3.6e) \quad \sum_{j \in \mathbf{q}_k} \bar{\mu}_k^j f_k^j(\hat{x}) \leq 0.$$

We claim that $\bar{\mu}_k = \hat{\mu}_k$. To obtain a contradiction, suppose that $\bar{\mu}_k \neq \hat{\mu}_k$. Because $\hat{\mu}_k^{j_k} = 1$, there must exist $j^* \neq j_k$ such that $\bar{\mu}_k^{j^*} > 0 = \hat{\mu}_k^{j^*}$. Since $k \in \hat{\mathbf{r}}(\hat{x})$ and $\psi(\hat{x}) = 0$, Lemma 3.6(c) implies that

$$(3.6f) \quad f_k^j(\hat{x}) \geq 0 \quad \text{for all } j \in \mathbf{q}_k.$$

Moreover, because $\hat{\mathbf{q}}_k(\hat{x}) = \{j_k\}$, $k \in \hat{\mathbf{r}}(\hat{x})$, and $\psi_k(\hat{x}) = 0$, we must have that $f_k^{j^*}(\hat{x}) > 0$. Hence, in view of (3.6f) and the fact that $\bar{\mu}_k^j \geq 0$ for all $j \in \mathbf{q}_k$, we have that

$$(3.6g) \quad \sum_{j \in \mathbf{q}_k} \bar{\mu}_k^j f_k^j(\hat{x}) = \bar{\mu}_k^{j^*} f_k^{j^*}(\hat{x}) + \sum_{\substack{j \in \hat{\mathbf{q}}_k \\ j \neq j^*}} \bar{\mu}_k^j f_k^j(\hat{x}) \geq \bar{\mu}_k^{j^*} f_k^{j^*}(\hat{x}) > 0,$$

which contradicts (3.6e). It follows that $\bar{\mu}_k = \hat{\mu}_k$, $k \in \hat{\mathbf{r}}(\hat{x})$, and hence $\mu_{k,i} \rightarrow \hat{\mu}_k$ as $i \rightarrow \infty$, $k \in \mathbf{r}$.

Therefore the sequence $\{(x_i, \mu_{1,i}, \mu_{2,i}, \dots, \mu_{r,i})\}_{i=i_0}^\infty$ is feasible for $\tilde{\mathbf{P}}$, converges to $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ as $i \rightarrow \infty$, and satisfies (3.6a) for all $i \geq i_0$. This implies that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is not a local minimizer for $(\tilde{\mathbf{P}})$, which proves the desired result. $\square$

ASSUMPTION 3.9. *Assumption 3.3 is satisfied, and for any $x \in \mathbb{R}^n$ such that $\psi(x) = 0$ we have the following:*

(a) *The vectors*

$$(3.7) \quad \{\nabla f_k^j(x)\}_{\substack{j \in \hat{\mathbf{q}}_k(x) \\ k \in \hat{\mathbf{r}}(x)}}$$

are linearly independent.

(b) *For any $k^* \in \hat{\mathbf{r}}(x)$,*

$$(3.8) \quad 0 \notin \text{co}_{\substack{j \in \hat{\mathbf{q}}_k(x) \\ k \in \hat{\mathbf{r}}(x) \setminus \{k^*\}}} \{\nabla f^0(x), \nabla f_k^j(x)\}.$$

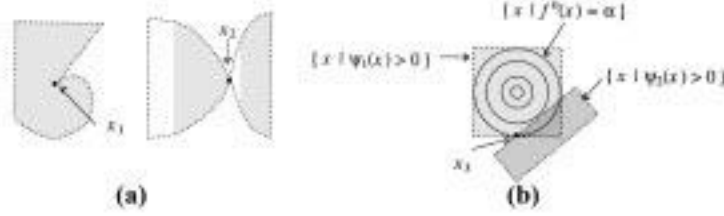


FIG. 3.1. (a) Assumption 3.9(a) fails at x_1 and x_2 . (b) Assumption 3.9(b) fails at x_3 . The constraint $\psi_2(x) \leq 0$ is redundant at x_3 since x_3 is also a local minimizer of $\min \{f^0(x) | \psi_1(x) \leq 0\}$.

Assumption 3.9(a), which requires linear independence of gradients of the active constraints, rules out certain pathological cases such as those depicted in Figure 3.1a, where the infeasible set is dotted. Assumption 3.9(b) is related to the strict complementary slackness condition. It implies that there are no “redundant” constraints at the boundary of the feasible set. In Figure 3.1b the point x_3 is a local minimizer for the problem $\min \{f^0(x) | \psi_1(x) \leq 0 \text{ and } \psi_2(x) \leq 0\}$. If we remove the constraint $\psi_2(x) \leq 0$, x_3 is also a local minimizer for the problem $\min \{f^0(x) | \psi_1(x) \leq 0\}$; i.e., the constraint $\psi_2(x) \leq 0$ is “redundant.” It is easily seen that Assumption 3.9(b) fails at x_3 .

To establish that $|\hat{\mathbf{q}}_k(\hat{x})| = 1$ for all $k \in \hat{\mathbf{r}}(\hat{x})$ whenever there exist $\hat{\mu}_k \in \Sigma_{q_k}$, $k \in \mathbf{r}$, such that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}})$ and $\psi(\hat{x}) = 0$, we will make use of the following result, whose proof is in the appendix.

Given $x \in \mathbb{R}^n$ and $\rho > 0$, we denote the closed ball of radius ρ around x by $B(x, \rho)$.

PROPOSITION 3.10. *Suppose that Assumption 3.9 is satisfied, that $\psi(\hat{x}) = 0$, and that there exist $\hat{\mu}_k \in \Sigma_{q_k}$, $k \in \mathbf{r}$, such that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}})$. Then there exist a $\hat{\rho} > 0$ and twice continuously differentiable functions $\tilde{f}^0: B(0, \hat{\rho}) \rightarrow \mathbb{R}$ and $\tilde{f}_k^j: B(0, \hat{\rho}) \rightarrow \mathbb{R}$, $k \in \mathbf{r}$, $j \in \mathbf{q}_k$, such that*

(a) $(0, \hat{\nu}_1, \dots, \hat{\nu}_r)$ is a local minimizer for the problem

$$(3.9a) \quad (\tilde{\mathbf{P}}_{\hat{x}}) \quad \min_{(z, \mu_1, \dots, \mu_r)} \left\{ \tilde{f}^0(z) \mid \sum_{j \in \mathbf{q}_k} \mu_k^j \tilde{f}_k^j(z) \leq 0, \mu_k \in \Sigma_{q_k}, k \in \mathbf{r}, z \in B(0, \hat{\rho}) \right\}$$

if and only if $(\hat{x}, \hat{\nu}_1, \dots, \hat{\nu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}})$.

(b) there exists a one-to-one map $l(\cdot, \cdot)$ taking pairs (k, j) , with $k \in \hat{\mathbf{r}}(\hat{x})$ and $j \in \hat{\mathbf{q}}_k(\hat{x})$, into $\{1, \dots, n\}$ and such that

$$(3.9b) \quad \tilde{f}_k^j(z) = z^{l(k,j)}, \quad k \in \hat{\mathbf{r}}(\hat{x}), \quad j \in \hat{\mathbf{q}}_k(\hat{x}),$$

for all $z \in B(0, \hat{\rho})$, where $z^{l(k,j)}$ denotes the $l(k, j)$ th component of z .

(c) $\partial \tilde{f}^0(0) / \partial z^s \leq 0$ if $s = l(k, j)$ for some $k \in \hat{\mathbf{r}}(\hat{x})$ and some $j \in \hat{\mathbf{q}}_k(\hat{x})$, and $\partial \tilde{f}^0(0) / \partial z^s = 0$ otherwise.

(d) For each $k \in \hat{\mathbf{r}}(\hat{x})$ and each $j \in \hat{\mathbf{q}}_k(\hat{x})$,

$$(3.9c) \quad \hat{\mu}_k^j \sum_{s \in \hat{\mathbf{q}}_k(\hat{x})} \left| \frac{\partial \tilde{f}^0}{\partial z^{l(k,s)}}(0) \right| = - \frac{\partial \tilde{f}^0}{\partial z^{l(k,j)}}(0).$$

(e) For each $k \in \hat{\mathbf{r}}(\hat{x})$, there exists a $j_k \in \hat{\mathbf{q}}(\hat{x})$ such that $\hat{\mu}^{k,j_k} > 0$, and

$$(3.9d) \quad \frac{\partial \tilde{f}^0}{\partial z^{l(k,j_k)}}(0) < 0.$$

THEOREM 3.11. *Suppose that Assumption 3.9 is satisfied, that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}})$, and that $\psi(\hat{x}) = 0$. Then $|\hat{\mathbf{q}}_k(\hat{x})| = 1$ for all $k \in \hat{\mathbf{r}}(\hat{x})$.*

Proof. Let $\hat{\rho} > 0$, $\tilde{f}^0: B(0, \hat{\rho}) \rightarrow \mathbb{R}$, $\tilde{f}_k^j: B(0, \hat{\rho}) \rightarrow \mathbb{R}$, $k \in \mathbf{r}$, $j \in \mathbf{q}_k$, $l(k, j)$, and $(\tilde{\mathbf{P}}_{\hat{x}})$ be as stipulated by Proposition 3.10. Since $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}})$, it follows from Proposition 3.10(a) that $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}}_{\hat{x}})$.

To obtain a contradiction, suppose that there exists $k^* \in \hat{\mathbf{r}}(\hat{x})$ such that $|\hat{\mathbf{q}}_{k^*}(\hat{x})| > 1$. We will show that this implies that $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is not a local minimizer of $(\tilde{\mathbf{P}}_{\hat{x}})$ by constructing a sequence $\{(z_i, \mu_{1,i}, \dots, \mu_{r,i})\}_{i=i_0}^\infty$, feasible for $(\tilde{\mathbf{P}}_{\hat{x}})$, converging to $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ as $i \rightarrow \infty$, and such that, for all $i \geq i_0$,

$$(3.10a) \quad \tilde{f}^0(z_i) < \tilde{f}^0(0).$$

By Proposition 3.10(e) there exists $j^* \in \hat{\mathbf{q}}_{k^*}(\hat{x})$ such that $\hat{\mu}_{k^*}^{j^*} > 0$ and $\partial \tilde{f}^0(0) / \partial z^{l(k^*, j^*)} < 0$. Next, since $|\hat{\mathbf{q}}_{k^*}(\hat{x})| > 1$ by assumption, there exists $j^{**} \neq j^*$ such that $j^{**} \in \hat{\mathbf{q}}_{k^*}(\hat{x})$. Let $\varepsilon > 0$ and $\nu \in \mathbb{R}^{q_{k^*}}$ be such that $\|\nu\| \leq \hat{\rho}$, $\langle \hat{\mu}_{k^*}, \nu \rangle = 0$, and $\nu^{j^*} - \nu^{j^{**}} = \varepsilon$. It is not difficult to show that such ε and ν exist. Since $k^* \in \hat{\mathbf{r}}(\hat{x})$, and $\psi(\hat{x}) = 0$, it follows from Corollary 3.7 that

$$(3.10b) \quad 0 = \langle \hat{\mu}_{k^*}, \nu \rangle = \sum_{j \in \mathbf{q}_{k^*}} \hat{\mu}_{k^*}^j \nu^j = \sum_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \hat{\mu}_{k^*}^j \nu^j.$$

For each $i \in \mathbb{N}$, let $z_i \in \mathbb{R}^n$ be defined by

$$(3.10c) \quad z_i^{l(k^*, j)} \triangleq \nu^j / i \quad \text{if } j \in \hat{\mathbf{q}}_{k^*}(\hat{x}) \setminus \{j^*\},$$

$$(3.10d) \quad z_i^{l(k^*, j^*)} \triangleq \nu^{j^*} / i + K / i^2,$$

$$(3.10e) \quad z_i^s \triangleq 0 \quad \text{if } s \notin \bigcup_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \{l(k^*, j)\},$$

where

$$(3.10f) \quad K \triangleq 2M \|\nu\|^2 / |\partial \tilde{f}^0(0) / \partial z^{l(k^*, j^*)}|,$$

with

$$(3.10g) \quad M \triangleq \max_{z \in B(0, \hat{\rho})} \|\tilde{f}_{zz}^0(z)\|.$$

Clearly, $z_i \rightarrow 0$ as $i \rightarrow \infty$, and there exists an i_1 such that for all $i \geq i_1$, $z_i \in B(0, \hat{\rho})$. Next, we construct the sequences $\{\mu_{k,i}\}$, $k \in \mathbf{r}$. We will consider two cases separately: $k \in \hat{\mathbf{r}}^c(\hat{x})$ and $k \in \hat{\mathbf{r}}(\hat{x})$.

Because $\psi(\hat{x}) = 0$, $\psi_k(\hat{x}) < 0$ for each $k \in \hat{\mathbf{r}}^c(\hat{x})$. Therefore, because $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is feasible for $(\tilde{\mathbf{P}})$, Lemma 3.4 implies that there exists an $i_2 \geq i_1$ and sequences $\{\mu_{k,i}\}_{i=i_2}^\infty \subset \Sigma_{q_k}$, $k \in \hat{\mathbf{r}}^c(\hat{x})$, such that $\mu_{k,i} \rightarrow \hat{\mu}_k$ as $i \rightarrow \infty$ and, for all $i \geq i_2$,

$$(3.10h) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j \tilde{f}_k^j(z_i) \leq 0, \quad k \in \hat{\mathbf{r}}^c(\hat{x}).$$

To complete the construction of the sequence $\{(z_i, \mu_{1,i}, \dots, \mu_{r,i})\}_{i=i_0}^\infty$, it remains to define the $\mu_{k,i}$'s for $k \in \hat{\mathbf{r}}(\hat{x})$. For each $k \in \hat{\mathbf{r}}(\hat{x}) \setminus \{k^*\}$, we set $\mu_{k,i} = \hat{\mu}_k$ for all i . It is obvious that $\mu_{k,i} \in \Sigma_{q_k}$ and that $\mu_{k,i} \rightarrow \hat{\mu}_k$ as $i \rightarrow \infty$. Moreover, using Corollary 3.7, (3.9b), and the fact that $z_i^{l(k,j)} = 0$ if $k \neq k^*$ (see (3.10e)), we get that, for each $k \in \hat{\mathbf{r}}(\hat{x}) \setminus \{k^*\}$,

$$(3.10i) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j \tilde{f}_k^j(z_i) = \sum_{j \in \mathbf{q}_k} \hat{\mu}_k^j \tilde{f}_k^j(z_i) = \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \hat{\mu}_k^j z_i^{l(k,j)} = 0.$$

Next, to construct $\{\mu_{k^*,i}\}_{i=i_0}^\infty$ we define, for each i ,

$$(3.10j) \quad \delta_i \triangleq \frac{K}{\varepsilon i} \hat{\mu}_{k^*}^{j^*}.$$

Because $\hat{\mu}_{k^*}^{j^*} > 0$ and $\hat{\mu}_{k^*} \in \Sigma_{q_{k^*}}$, we must have $\hat{\mu}_{k^*}^{j^{**}} < 1$. Since $\delta_i \rightarrow 0$ as $i \rightarrow \infty$, there exists an $i_3 \geq i_2$ such that for all $i \geq i_3$,

$$(3.10k) \quad \delta_i \leq \min\{\hat{\mu}_{k^*}^{j^*}, 1 - \hat{\mu}_{k^*}^{j^{**}}\}.$$

For all $i \geq i_3$, let $\mu_{k^*,i}$ be defined by

$$(3.10l) \quad \mu_{k^*,i}^{j^*} \triangleq \hat{\mu}_{k^*}^{j^*} - \delta_i,$$

$$(3.10m) \quad \mu_{k^*,i}^{j^{**}} \triangleq \hat{\mu}_{k^*}^{j^{**}} + \delta_i,$$

$$(3.10n) \quad \mu_{k^*,i}^j \triangleq \hat{\mu}_{k^*}^j \quad \text{for all } j \in \mathbf{q}_{k^*} \setminus \{j^*, j^{**}\}.$$

Then $\mu_{k^*,i} \rightarrow \hat{\mu}_{k^*}$ as $i \rightarrow \infty$ and, because of (3.10k), $\mu_{k^*,i} \in \Sigma_{q_{k^*}}$ for all $i \geq i_3$. Moreover, in view of the fact that $\{j^*, j^{**}\} \subset \hat{\mathbf{q}}_{k^*}(\hat{x})$, and using Corollary 3.7, (3.10c–e), (3.10l–n), and Proposition 3.10(b), we have, for all $i \geq i_3$, that

$$\begin{aligned} \sum_{j \in \mathbf{q}_{k^*}} \mu_{k^*,i}^j \tilde{f}_{k^*}^j(z_i) &= \sum_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \mu_{k^*,i}^j \tilde{f}_{k^*}^j(z_i) = \sum_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \mu_{k^*,i}^j z_i^{l(k^*,j)} \\ &= \left(\sum_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x}) \setminus \{j^*, j^{**}\}} \hat{\mu}_{k^*}^j \nu^j / i \right) + (\hat{\mu}_{k^*}^{j^{**}} + \delta_i) \nu^{j^{**}} / i \\ &\quad + (\hat{\mu}_{k^*}^{j^*} - \delta_i) (\nu^{j^*} / i + K / i^2) \\ (3.10o) \quad &= \sum_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \hat{\mu}_{k^*}^j \nu^j / i + \delta_i / i (\nu^{j^{**}} - \nu^{j^*} - K / i). \end{aligned}$$

Since ν was chosen so that $\nu^{j^*} - \nu^{j^{**}} = \varepsilon$, (3.10b) and (3.10o) imply that

$$(3.10p) \quad \sum_{j \in \mathbf{q}_{k^*}} \mu_{k^*,i}^j \tilde{f}_{k^*}^j(z_i) = \frac{\delta_i}{i} (-\varepsilon - K / i) \leq 0.$$

It follows from (3.10h), (3.10i), and (3.10p) that $\{(z_i, \mu_{1,i}, \dots, \mu_{r,i})\}_{i=i_3}^\infty$ is feasible for $(\tilde{\mathbf{P}}_{\hat{x}})$.

Next, from (3.10g) we get that, for each $i \geq i_3$,

$$(3.10q) \quad \tilde{f}^0(z_i) - \tilde{f}^0(0) \leq \sum_{j \in \mathbf{n}} \frac{\partial \tilde{f}^0(0)}{\partial z^j} z_i^j + \frac{1}{2} M \|z_i\|^2.$$

In view of (3.10c–e), (3.10q) implies that

$$(3.10r) \quad \begin{aligned} \tilde{f}^0(z_i) - \tilde{f}^0(0) &\leq \sum_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \frac{\partial \tilde{f}^0(0)}{\partial z^{l(k^*,j)}} z_k^{l(k^*,j)} + \frac{1}{2} M \|z_i\|^2 \\ &\leq \frac{1}{i} \sum_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \frac{\partial \tilde{f}^0(0)}{\partial z^{l(k^*,j)}} \nu^j + \frac{K}{i^2} \frac{\partial \tilde{f}^0(0)}{\partial z^{l(k^*,j^*)}} + \frac{1}{2} M \|z_i\|^2. \end{aligned}$$

Since for any vectors $a, b \in \mathbb{R}^n$ we have that $\|a + b\|^2 \leq 2(\|a\|^2 + \|b\|^2)$, it follows from (3.10c–e) that, for each i ,

$$(3.10s) \quad \|z_i\|^2 \leq \frac{2}{i^2} (\|\nu\|^2 + K^2/i^2).$$

In view of (3.10s), (3.10c), and (3.10b), (3.10r) implies that

$$(3.10t) \quad \begin{aligned} \tilde{f}^0(z_i) - \tilde{f}^0(0) &\leq -\frac{1}{i} \left(\sum_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \hat{\mu}_{k^*}^j \nu^j \right) \sum_{s \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \left| \frac{\partial \tilde{f}^0(0)}{\partial z^{l(k^*,s)}} \right| \\ &\quad + \frac{K}{i^2} \frac{\partial \tilde{f}^0(0)}{\partial z^{l(k^*,j^*)}} + \frac{M}{i^2} \left(\|\nu\|^2 + \frac{K^2}{i^2} \right) \\ &\leq \left(K \frac{\partial \tilde{f}^0(0)}{\partial z^{l(k^*,j^*)}} + M(\|\nu\|^2 + K^2/i^2) \right) / i^2. \end{aligned}$$

Finally, in view of (3.10f) and because $\partial \tilde{f}^0(0)/\partial z^{l(k^*,j^*)} < 0$, (3.10t) implies that there exists an $i_0 \geq i_3$ such that, for all $i \geq i_0$,

$$(3.10u) \quad \tilde{f}^0(z_i) - \tilde{f}^0(0) \leq M(-\|\nu\|^2 + K^2/i^2)/i^2 < 0.$$

Together with (3.10h), (3.10i), and (3.10p), (3.10u) implies that $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is not a local minimizer for $(\tilde{\mathbf{P}})$, which is a contradiction.

Using Theorems 3.2 and 3.11 and Proposition 3.5, we obtain the following result.

THEOREM 3.12. *Suppose that Assumption 3.9 is satisfied. Then $\hat{x} \in \mathbb{R}^n$ is a local minimizer for $\mathbf{P}$ if and only if there exists $\hat{\mu}_k \in \Sigma_{q_k}$, $k \in \mathbf{r}$, such that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}})$.*

4. Numerical examples. In this section, we present numerical examples that show how the transformation proposed can be used to solve optimization problems with max-min constraints. We divide this section into two subsections: the first contains examples of optimization problems with max-min constraints in $\mathbb{R}^2$, while the second contains examples of problems obtained by discretizing optimal control problems.

4.1. Optimization problems with max-min constraints in $\mathbb{R}^2$. In this subsection, we construct test problems by adding a max-min constraint to two unconstrained optimization test problems from [MoG.1], namely, the Rosenbrock test function and the Beale test function.

To obtain a parametrized family of max-min constraints, we proceed as follows. Given parameters $a \in \mathbb{R}^2$ and $b > 0$, let

$$(4.1a) \quad h_1^1(x, a, b) \triangleq 1.5b + \frac{(x^1 - a^1)^2}{2b} - (x^2 - a^2),$$

$$(4.1b) \quad h_1^2(x, a, b) \triangleq -0.5b - \frac{(x^1 - a^1)^2}{2b} - (x^2 - a^2),$$

$$(4.1c) \quad h_1^3(x, a, b) \triangleq 1.5b - \frac{(x^2 - a^2 - b)^2}{2b} - (x^1 - a^1),$$

$$(4.1d) \quad h_1^4(x, a, b) \triangleq 1.5b - \frac{(x^2 - a^2 - b)^2}{2b} + (x^1 - a^1),$$

$$(4.1e) \quad h_2^1(x, a, b) \triangleq b + (x^2 - a^2) + \frac{(x^1 - a^1)^2}{b},$$

$$(4.1f) \quad h_2^2(x, a, b) \triangleq 2b - (x^2 - a^2) - 2(x^1 - a^1) - \frac{2}{b}(x^1 - a^1)^2,$$

$$(4.1g) \quad h_2^3(x, a, b) \triangleq 2b - (x^2 - a^2) + 2(x^1 - a^1) - \frac{2}{b}(x^1 - a^1)^2,$$

and set $q_1 \triangleq 4$, $q_2 \triangleq 3$. The max-min constraints considered in this subsection will have the form

$$(4.1h) \quad \max_{k \in \{1, 2\}} \min_{j \in \mathbf{q}_k} h_k^j(x, a, b) \leq 0.$$

Note that the vector $x = a$ does not satisfy the inequality (4.1h).

Given an objective function, we construct a test problem by choosing a so that the unconstrained minimizer of that function does not satisfy the max-min constraints (4.1h). The parameter b is used to control the size of the infeasible region created by the max-min constraints. For $a = 0$, (4.1h) above is satisfied if and only if $x \in \mathbb{R}^2$ is not strictly inside the shaded infeasible region shown in Figure 4.1a.

Rosenbrock function. The objective function for our first example is the Rosenbrock function

$$(4.2a) \quad f_R^0(x) \triangleq 100(x^2 - (x^1)^2)^2 + (1 - x^1)^2,$$

which has a global minimizer at $a_R = (1, 1)$. For $r = 2$, $q_1 = 4$, and $q_2 = 3$, we define $f_k^j: \mathbb{R}^2 \rightarrow \mathbb{R}$, $k \in \mathbf{r}$, $j \in \mathbf{q}_k$, by

$$(4.2b) \quad f_k^j(x) \triangleq h_k^j(x, a_R, 0.5),$$

where the functions $h_k^j(\cdot, \cdot, \cdot)$ are given by (4.1a–g).

To solve the problem

$$(4.2c) \quad (\text{P}_R) \quad \min_x \{f_R^0(x) \mid \max_{k \in \mathbf{r}} \min_{j \in \mathbf{q}_k} f_k^j(x) \leq 0\},$$

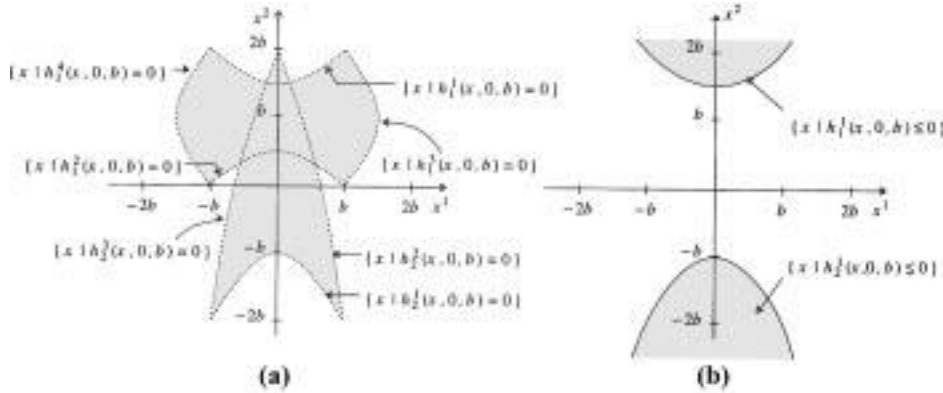


FIG. 4.1. Geometry of the infeasible set for $a = 0$. (a) The infeasible set. (b) The sets $\{x | h_1^1(x, 0, b) \leq 0\}$ and $\{x | h_2^1(x, 0, b) \leq 0\}$ do not intersect.

we define the equivalent problem

$$(4.2d) \quad (\tilde{P}_R) \quad \min_{(x, \mu_1, \mu_2)} \left\{ f_R^0(x) \mid \sum_{j \in \mathbf{q}_k} \mu_k^j f_k^j(x) \leq 0, \mu_k \in \Sigma_{q_k}, k \in \mathbf{r} \right\}.$$

Problem $(\tilde{P}_R)$ can be solved using nonlinear programming software such as NPSOL [GiM.1]. To evaluate the reformulation proposed, we compare the computational work needed for NPSOL to solve $(\tilde{P}_R)$ with the work needed to solve (P_R) using the decomposition strategy explained in the introduction. Hence, we compare the work needed to solve $(\tilde{P}_R)$ with that needed to solve

$$(4.2e) \quad (P_{R,j_1,j_2}) \quad \min_x \{f_R^0(x) \mid f_1^{j_1}(x) \leq 0, f_2^{j_2}(x) \leq 0\}$$

for all $j_1 \in \mathbf{q}_1$ and $j_2 \in \mathbf{q}_2$, a total of 12 problems, which can also be solved numerically using NPSOL.

We will measure the total work required to solve a problem by counting the number of evaluations of the functions $f_R^0(\cdot)$, $f_k^j(\cdot)$ (denoted by nf_R^0 and nf_k^j) and of their respective gradients (denoted by $n\nabla f_R^0$ and $n\nabla f_k^j$). When solving $(\tilde{P}_R)$, we used

$$(4.2f) \quad \mu_{k,0}^j = 1/q_k, \quad k \in \mathbf{r}, \quad j \in \mathbf{q}_k,$$

as the initial guess for the μ_k 's. The results obtained for an initial guess $x_0 = (-1, 1)$ are shown in Table 4.1(a), while the results corresponding to an initial guess $x_0 = (2, 2)$ are in Table 4.1(b). In both tables, the number of function and gradient evaluations given in the rows labeled (P_{R,j_1,j_2}) correspond to the sum of the number of function and gradient evaluations needed to solve all the problems (P_{R,j_1,j_2}) , with $j_1 \in \mathbf{q}_1$ and $j_2 \in \mathbf{q}_2$. The problems (P_{R,j_1,j_2}) were all solved using the same initial guess x_0 .

Solving $(\tilde{P}_R)$, using NPSOL, with $x_0 = (-1, 1)$ and $\mu_{k,0}$ as in (4.2f), after 30 iterations we obtained the local minimizer $\hat{x} = (0.809416, 0.655609)$ (up to six digits), which corresponds to the objective function value of 0.0363430. This point is also computed when $(P_{R,3,3})$ is solved with NPSOL starting from $x_0 = (-1, 1)$. If we instead set $x_0 = (2, 2)$, NPSOL computes, after 21 iterations, the local minimizer $\hat{x} =$

TABLE 4.1(a)
Rosenbrock function; $x_0 = (-1, 1)$.

	nf_R^0	$n\nabla f_R^0$	nf_k^j	$n\nabla f_k^j$
$(\tilde{P}_R)$	41	40	376	246
(P_{R,j_1,j_2})	376	347	694	670

TABLE 4.1(b)
Rosenbrock function; $x_0 = (2, 2)$.

	nf_R^0	$n\nabla f_R^0$	nf_k^j	$n\nabla f_k^j$
$(\tilde{P}_R)$	31	30	186	180
(P_{R,j_1,j_2})	351	340	702	678

TABLE 4.2(a)
Beale function; $x_0 = (1, 1)$.

	nf_B^0	$n\nabla f_B^0$	nf_k^j	$n\nabla f_k^j$
$(\tilde{P}_B)$	21	20	126	120
(P_{B,j_1,j_2})	273	249	546	522

TABLE 4.2(b)
Beale function; $x_0 = (1, -1)$.

	nf_B^0	$n\nabla f_B^0$	nf_k^j	$n\nabla f_k^j$
$(\tilde{P}_B)$	21	20	126	120
(P_{B,j_1,j_2})	227	215	454	430

(1.171476, 1.373510), which corresponds to an objective function value of 0.0029532. NPSOL computes the same point when it is used to solve $(P_{R,2,3})$ starting from $x_0 = (2, 2)$.

For both initial conditions, $x_0 = (-1, 1)$ and $x_0 = (2, 2)$, NPSOL failed to solve $(P_{R,1,1})$, which has an empty feasible set (see Figure 4.1b).

Beale function. We ran a set of experiments similar to the above using the Beale function [MoG.1] as the objective function. It is defined by

(4.3a)

$$f_B^0(x) \triangleq (1.5 - x^1(1 - x^2))^2 + (2.25 - x^1(1 - (x^2)^2))^2 + (2.625 - x^1(1 - (x^2)^3))^2.$$

The function $f_B^0(\cdot)$ has an unconstrained minimum at $a_B = (3, 0.5)$, which corresponds to a value of the objective function of 0. With $a = 0.2$, $r = 2$, $q_1 = 4$, and $q_2 = 3$, we set

$$(4.3b) \quad f_k^j(x) \triangleq h_k^j(x, a_B, 0.2), \quad k \in \mathbf{r}, \quad j \in \mathbf{q}_k,$$

and define the problems $(\tilde{P}_B)$ and the corresponding (P_{B,j_1,j_2}) , $j_1 \in \mathbf{q}_1$, $j_2 \in \mathbf{q}_2$, following the pattern set above. We compare the amount of work needed for NPSOL to solve $(\tilde{P}_B)$ and the (P_{B,j_1,j_2}) from two distinct initial conditions, $x_0 = (1, 1)$ and $x_0 = (1, -1)$. When solving (P_B) the initial guess for the μ_k 's is taken as in (4.2f). The number of function and gradient evaluations for the two initial guesses are shown in Tables 4.2(a) and 4.2(b). Again, the numbers given in the rows labeled (P_{B,j_1,j_2}) correspond to the sum of the number of function and gradient evaluations needed to solve all (P_{B,j_1,j_2}) with $j_1 \in \mathbf{q}_1$ and $j_2 \in \mathbf{q}_2$.

For both initial conditions, when solving $(\tilde{P}_B)$, NPSOL converged to the local minimizer $\hat{x} = (2.670861, 2.04083937)$. The same point is obtained when we use NPSOL to solve $(P_{B,2,3})$, starting from either initial condition. This point corresponds to a value of the objective function of 0.0255337. As expected, NPSOL fails to converge when it is used to solve $(P_{1,1})$, which has an empty feasible set.

The results from Tables 4.1 and 4.2 indicate that solving the modified problem $(\tilde{P})$, obtained by applying the transcription proposed in section 2, is significantly more efficient than the decomposition strategy explained in the introduction. We believe that this difference becomes even greater when the number of min constraints r increases.

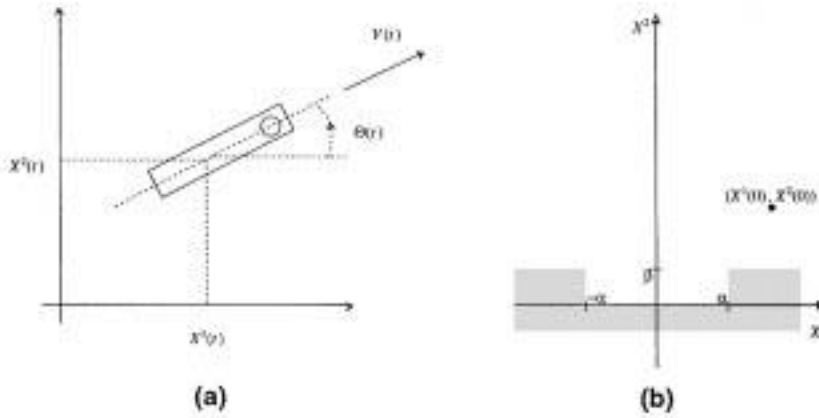


FIG. 4.2. (a) Wheeled robot: the circle indicates the front of the robot. (b) The shaded area is the obstacle to be avoided by the robot.

4.2. Optimal control in the presence of obstacles. In this subsection, we apply the transformation discussed in sections 2 and 3 to solve an optimal control problem with max-min constraints. Problems of this kind with a rather large number of min constraints are generated by discretizing optimal control problems related to robotic path-planning in the presence of obstacles.

Problem statement. We consider a wheeled mobile robot whose dynamics are similar to that of a unicycle [ReS.1]. At time t , the state of this system is given by its coordinates on the plane $X^1(t)$ and $X^2(t)$, angular orientation $\Theta(t)$, velocity $V(t)$, and angular velocity $\Omega(t)$. We suppose that we can control the acceleration $U^1(t)$ and the angular acceleration $U^2(t)$. If its wheels do not slide, then the robot is modeled by the following equations (see Figure 4.2a):

$$(4.4a) \quad \frac{d}{dt}X^1(t) = V(t) \cos(\Theta(t)),$$

$$(4.4b) \quad \frac{d}{dt}X^2(t) = V(t) \sin(\Theta(t)),$$

$$(4.4c) \quad \frac{d}{dt}\Theta(t) = \Omega(t),$$

$$(4.4d) \quad \frac{d}{dt}V(t) = U^1(t),$$

$$(4.4e) \quad \frac{d}{dt}\Omega(t) = U^2(t),$$

for all $t \in \mathbb{R}$ and with initial condition $(X^1(0), X^2(0), \Theta(0), V(0), \Omega(0))$.

We seek controls $U^1(\cdot), U^2(\cdot)$ that steer the dynamical system (4.4a–e) in minimum time T from a given initial condition $(X^1(0), X^2(0), \Theta(0), 0, 0)$ to the state $(0, 0, 0, 0, 0)$, while satisfying the constraints

$$\begin{aligned}
(4.5a) \quad & X^2(t) \geq 0 \quad \text{for all } t \in [0, T], \\
(4.5b) \quad & X^1(t) \leq \alpha \quad \text{or} \quad X^2(t) \geq \beta \quad \text{for all } t \in [0, T], \\
(4.5c) \quad & X^1(t) \geq -\alpha \quad \text{or} \quad X^2(t) \geq \beta \quad \text{for all } t \in [0, T], \\
(4.5d) \quad & V(t) \in [-1, 1] \quad \text{for all } t \in [0, T], \\
(4.5e) \quad & U^1(t) \in [-2, 2], \quad U^2(t) \in [-\pi/4, \pi/4] \quad \text{for all } t \in [0, T],
\end{aligned}$$

where α and β are given positive constants. The constraints (4.5a–c) limit the area of the $X^1 - X^2$ plane where the robot can travel (see Figure 4.2b), (4.5d) bounds the velocity, and (4.5e) bounds the acceleration and angular acceleration.

Since T is unknown a priori, we introduce a new variable $\tau \triangleq t/T$ taking values in $[0, 1]$, and define, for every $\tau \in [0, 1]$, $x^1(\tau) \triangleq X^1(\tau T)$, $x^2(\tau) \triangleq X^2(\tau T)$, $\theta(\tau) \triangleq \Theta(\tau T)$, $\omega(\tau) \triangleq \Omega(\tau T)$, $\nu(\tau) \triangleq V(\tau T)$, $u^1(\tau) = U^1(\tau T)$, and $u^2(\tau) = U^2(\tau T)$. In terms of these variables, (4.4a–e) can be rewritten as

$$\begin{aligned}
(4.6a) \quad & \dot{x}^1(\tau) = T\nu(\tau) \cos(\theta(\tau)), \quad \tau \in [0, 1], \\
(4.6b) \quad & \dot{x}^2(\tau) = T\nu(\tau) \sin(\theta(\tau)), \quad \tau \in [0, 1], \\
(4.6c) \quad & \dot{\theta}(\tau) = T\omega(\tau), \quad \tau \in [0, 1], \\
(4.6d) \quad & \dot{\nu}(\tau) = Tu^1(\tau), \quad \tau \in [0, 1], \\
(4.6e) \quad & \dot{\omega}(\tau) = Tu^2(\tau), \quad \tau \in [0, 1],
\end{aligned}$$

where the dot denotes differentiation with respect to τ and with initial conditions

$$(4.6f) \quad (x^1(0), x^2(0), \theta(0), \nu(0), \omega(0)) = (X^1(0), X^2(0), \Theta(0), V(0), \Omega(0)).$$

We define

$$(4.7a) \quad C \triangleq \{(T, u) \in \mathbb{R} \times L_2^2[0, 1] \mid T \geq 0, u^1(\tau) \in [-2, 2], u^2(\tau) \in [-\pi/4, \pi/4] \text{ for a.e. } \tau \in [0, 1]\},$$

where $L_2^2[0, 1]$ denotes the space of square integrable functions on $[0, 1]$ taking values in $\mathbb{R}^2$. Given an element $\eta = (T, u)$ of $\mathcal{C}$, we denote the corresponding solutions of (4.6a–e) by $x^{1,\eta}, x^{2,\eta}, \theta^\eta, \nu^\eta$, and ω^η . Finally, we define the constraint functions $\psi_k: \mathcal{C} \rightarrow \mathbb{R}$, $k \in \{1, 2\}$, by

$$(4.7b) \quad \psi_k(\eta) = \max_{\tau \in [0, 1]} \phi_k(\eta, \tau),$$

where the functions $\phi_k: \mathcal{C} \times [0, 1] \rightarrow \mathbb{R}$, $k \in \{1, 2\}$, are defined by

$$(4.7c) \quad \phi_1(\eta, \tau) \triangleq \min\{x^{1,\eta}(\tau) - \alpha, \beta - x^{2,\eta}(\tau)\},$$

$$(4.7d) \quad \phi_2(\eta, \tau) \triangleq \min\{-x^{1,\eta}(\tau) - \alpha, \beta - x^{2,\eta}(\tau)\}.$$

The minimum time optimal control problems stated above can be formulated as follows:

$$\begin{aligned}
(\text{P}_{oc}) \quad & \min_{\eta \in \mathcal{C}} \{T \mid \max_{1 \leq k \leq 2} \psi_k(\eta) \leq 0, \max_{\tau \in [0, 1]} \max\{|\nu^\eta(\tau)| - 1, -x^{2,\eta}(\tau)\} \leq 0, \\
& (x^{1,\eta}(1), x^{2,\eta}(1), \theta^\eta(1), \nu^\eta(1), \omega^\eta(1)) = (0, 0, 0, 0, 0)\}.
\end{aligned}$$

Discretization. To solve (P_{oc}) numerically, we discretize it following [Pol.2, Pol.3, KiP.1] and obtain a finite-dimensional mathematical programming problem whose solution is in a one-to-one correspondence with an approximate solution of (P_{oc}) .

Given $N > 0$, let $\mathcal{T}_N$ be a mesh on $[0,1]$ with N intervals and mesh points $\{\tau_0, \tau_1, \dots, \tau_N\}$, with $\tau_s < \tau_{s+1}$, $i \in \{0, 1, \dots, N-1\}$, $\tau_0 = 0$, and $\tau_N = 1$. Let

$$(4.8a) \quad \mathcal{P}_N \triangleq \{(T, \nu) \in \mathbb{R} \times L_2^2[0,1] \mid \nu^1, \nu^2 \text{ are piecewise linear on the mesh } \mathcal{T}_N\}.$$

We define the discrete set of admissible controls by $\mathcal{C}_N \triangleq \mathcal{C} \cap \mathcal{P}_N$. Note that $\mathcal{P}_N$ has dimension $2N+1$, and hence $\mathcal{C}_N \subset \mathcal{P}_N$ is in one-to-one correspondence with a subset of $\mathbb{R}^{2N+1}$.

Given $\eta = (T, u) \in \mathcal{C}_N$, we let $x_N^{1,\eta}, x_N^{2,\eta}, \theta_N^\eta, \nu_N^\eta$, and ω_N^η —all elements of $\mathbb{R}^{N+1}$ —denote the solutions of the following difference equations:

$$(4.8b) \quad x_N^1(s+1) - x_N^1(s) = T\nu_N(s) \cos(\theta_N(s))(\tau_{s+1} - \tau_s), \quad 0 \leq s \leq N-1,$$

$$(4.8c) \quad x_N^2(s+1) - x_N^2(s) = T\nu_N(s) \sin(\theta_N(s))(\tau_{s+1} - \tau_s), \quad 0 \leq s \leq N-1,$$

$$(4.8d) \quad \theta_N(s+1) - \theta_N(s) = T\omega_N(s)(\tau_{s+1} - \tau_s), \quad 0 \leq s \leq N-1,$$

$$(4.8e) \quad \nu_N(s+1) - \nu_N(s) = Tu^1(s)(\tau_{s+1} - \tau_s), \quad 0 \leq s \leq N-1,$$

$$(4.8f) \quad \omega_N(s+1) - \omega_N(s) = Tu^2(s)(\tau_{s+1} - \tau_s), \quad 0 \leq s \leq N-1,$$

with $(x_N^1(0), x_N^2(0), \theta_N(0), \nu_N(0), \omega_N(0)) = (x^1(0), x^2(0), \theta(0), \nu(0), \omega(0))$. Equations (4.8b–f) are obtained by applying Euler's forward method to (4.6a–e).

Next, to obtain discrete counterparts for the max-min constraints, we define the functions $\psi_{k,N}: \mathcal{C}_N \rightarrow \mathbb{R}$, $k \in \{1, 2\}$, by

$$(4.8g) \quad \psi_{k,N}(\eta) = \max_{0 \leq s \leq N} \phi_{k,N}(\eta, s),$$

where $\phi_{k,N}: \mathcal{C}_N \times \{0, 1, \dots, N\} \rightarrow \mathbb{R}$, $k \in \{1, 2\}$, are defined by

$$(4.8h) \quad \phi_{1,N}(\eta, s) \triangleq \min\{x_N^{1,\eta}(s) - \alpha, \beta - x_N^{2,\eta}(s)\},$$

$$(4.8i) \quad \phi_{2,N}(\eta, s) \triangleq \min\{-x_N^{1,\eta}(s) - \alpha, \beta - x_N^{2,\eta}(s)\}.$$

We replace the requirement that the control steers the system to the origin by the inequality constraint

$$(4.8j) \quad \|(x_N^{1,\eta}(1), x_N^{2,\eta}(1), \theta_N^\eta(1), \nu_N^\eta(1), \omega_N^\eta(1))\|_\infty \leq C/N,$$

where $C > 0$ is a fixed constant (e.g., 10^{-6}). Hence, we obtain the following approximating (see [Pol.2]) finite-dimensional optimization problem

$$(P_{oc,N}) \quad \min_{\eta \in \mathcal{C}_N} \{T \mid \max_{1 \leq k \leq 2} \psi_{k,N}(\eta) \leq 0, \max_{0 \leq s \leq N} \max\{|\nu_N^\eta(s)| - 1, -x_{2,N}^\eta(s)\} \leq 0,$$

$$\|(x_N^{1,\eta}(1), x_N^{2,\eta}(1), \theta_N^\eta(1), \nu_N^\eta(1), \omega_N^\eta(1))\|_\infty \leq C/N\}.$$

The constraint

$$(4.9a) \quad \max_{1 \leq k \leq 2} \psi_{k,N}(\eta) \leq 0$$

corresponds to the requirements that

$$(4.9b) \quad \max_{0 \leq s \leq N} \min\{x_N^{1,\eta}(s) - \alpha, \beta - x_N^{2,\eta}(s)\} \leq 0,$$

$$(4.9c) \quad \max_{0 \leq s \leq N} \min\{-x_N^{1,\eta}(s) - \alpha, \beta - x_N^{2,\eta}(s)\} \leq 0,$$

a total of $2N + 2$ min constraints. Since each min is taken between two functions, if we were to follow the decomposition strategy explained in the introduction, we would have to solve 2^{2N+2} optimal control problems. Typical values for N range between 20 and 200, depending on the problem. Hence, for $N = 100$ we would have to solve 2^{202} (approximately 10^{60}) optimal control problems, which is not a very attractive prospect.

To solve $(P_{oc,N})$ we use our reformulation as in sections 2 and 3. To do that, we introduce the variables $\mu_{k,N} = (\mu_{k,N}(0), \mu_{k,N}(1), \dots, \mu_{k,N}(N)) \in (\Sigma_2)^{N+1}$, where $\mu_{k,N}(s) = (\mu_{k,N}^1(s), \mu_{k,N}^2(s))$, $k \in \{1, 2\}$. We also define the functions $\xi_{k,N}: \mathcal{C}_N \times (\Sigma_2)^{N+1} \times \{0, 1, \dots, N\} \rightarrow \mathbb{R}$ by

$$(4.9d) \quad \xi_{1,N}(\eta, \mu_{1,N}, s) \triangleq \mu_{1,N}^1(s)(x_N^{1,\eta}(s) - \alpha) + \mu_{1,N}^2(s)(\beta - x_N^{2,\eta}(s)),$$

$$(4.9e) \quad \xi_{2,N}(\eta, \mu_{2,N}, s) \triangleq \mu_{2,N}^1(s)(-x_N^{1,\eta}(s) - \alpha) + \mu_{2,N}^2(s)(\beta - x_N^{2,\eta}(s)).$$

The discrete optimal control problem $(\tilde{P}_{oc,N})$ is defined as follows.

$$\begin{aligned} (\tilde{P}_{oc,N}) \quad & \min_{(\eta, \mu_{1,N}, \mu_{2,N})} \{T | \eta \in \mathcal{C}_N, \mu_{k,N} \in (\Sigma_2)^{N+1}, k \in \{1, 2\}, \\ & \max_{1 \leq k \leq 2} \max_{0 \leq s \leq N} \xi_{k,N}(\eta, \mu_{k,N}, s) \leq 0, \\ & \|(x_N^{1,\eta}(1), x_N^{2,\eta}(1), \theta_N^\eta(1), \nu_N^\eta(1), \omega_N^\eta(1))\|_\infty \leq C/N, \\ & \max_{0 \leq s \leq N} \max\{|\nu_N^\eta(s)| - 1, -x_{2,N}^\eta(s)\} \leq 0\}. \end{aligned}$$

Problem $(\tilde{P}_{oc,N})$ can be easily rewritten as a nonlinear programming problem with smooth inequality constraints. Therefore, it can be solved using NPSOL.

Consider for example the case where obstacle is defined by $\alpha = 0.25$, $\beta = 0.2$ (see Figure 4.2b), and the initial condition is $x^1(0) = 0.3$, $x^2(0) = 0.25$, $\theta(0) = 0$, $\nu(0) = 0$, $w(0) = 0$. The computed optimal trajectory is shown in Figure 4.3, where the dashed line corresponds to the optimal trajectory when the max-min constraints are not present; i.e., the obstacles are removed (but the constraints on the velocity, acceleration, and the requirement that $x^2(t) \geq 0$ for all t are enforced), and the full line corresponds to the optimal trajectory with the max-min constraints. The obstacles to be avoided are also shown for reference. Figure 4.4 shows the computed controls u^1 and u^2 , velocity and angular velocity.

Without the max-min constraints, the computed minimum time is 2.8910. When the max-min constraints are enforced, the minimum time is 3.0707.

The final mesh for solving the example without the max-min constraints was uniform and had 80 points. The points making up the meshes used for solving $(\tilde{P}_{oc,N})$ were chosen using an adaptive scheme as in [MaP.2]. The final number of mesh points when max-min constraints were present was 108. In both cases, i.e., with and without the max-min constraints, a diagonalization strategy (see [HeP.1]) was used to increase the number of mesh points N from 10 to the final value.

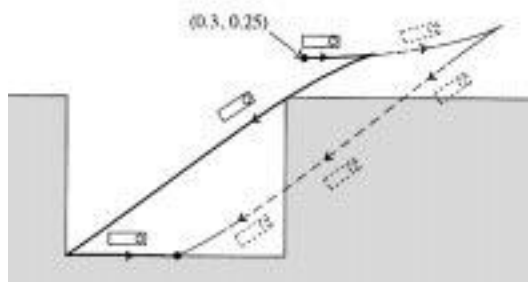


FIG. 4.3. Computed optimal trajectories: the dashed line shows the optimal trajectory with no max-min constraints (but the other constraints enforced). The full line shows the trajectory with max-min constraints.

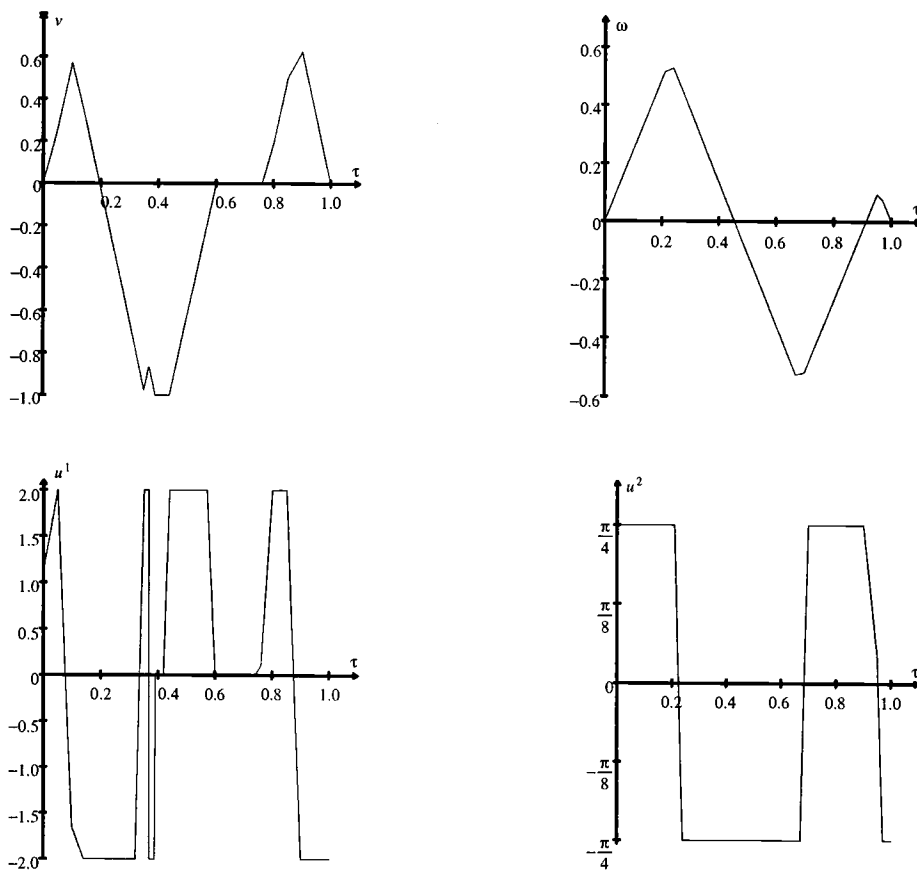


FIG. 4.4. Optimal controls and corresponding velocity and angular velocity.

5. Conclusion. We have presented a transcription that allows one to transform an optimization problem with max-min constraints into a single inequality constrained nonlinear program with smooth constraints. We have shown that the global and local minimizers of the problem with the max-min constraints correspond to those of the problem obtained via the proposed transformation and vice versa.

The major advantage of using our transformation is that it eliminates the need for solving an optimization problem with max-min constraints by solving a potentially very large collection of “ordinary” nonlinear programming problems. Our numerical results indicate that, in solving even relatively simple optimization problems with max-min constraints, the computing time savings can be very considerable. Our results also show that our approach works very well even for problems with a very large number of max-min constraints, such as those generated by discretizing certain optimal control problems.

Appendix. In this appendix we present the proof of Proposition 3.10. It is divided into several steps. First, we construct the functions $\tilde{f}^0(\cdot)$, $\tilde{f}_k^j(\cdot)$, $k \in \mathbf{r}$, $j \in \mathbf{q}_k$, using the implicit function theorem. Then we show that parts (a)–(e) of Proposition 3.10 hold.

Throughout this appendix we assume that Assumption 3.9 is satisfied and that $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer of $(\tilde{P})$ such that $\psi(\hat{x}) = 0$. We also assume, without loss of generality (one can always re-index the functions $f_k^j(\cdot)$), that $\hat{\mathbf{r}}(\hat{x}) = \{1, \dots, m\}$ and $\hat{\mathbf{q}}_k(\hat{x}) = \{1, \dots, p_k\}$, $k \in \hat{\mathbf{r}}(\hat{x})$.

We denote

$$(A.1a) \quad \bar{p} \triangleq \sum_{k \in \hat{\mathbf{r}}(\hat{x})} |\hat{\mathbf{q}}_k(\hat{x})| = \sum_{k \in \hat{\mathbf{r}}(\hat{x})} p_k$$

and let the functions $f_k: \mathbb{R}^n \rightarrow \mathbb{R}^{p_k}$, $k \in \hat{\mathbf{r}}(\hat{x})$, and $\bar{f}: \mathbb{R}^n \rightarrow \mathbb{R}^{\bar{p}}$ be defined by

$$(A.1b) \quad f_k(x) \triangleq [f_k^1(x) f_k^2(x) \cdots f_k^{p_k}(x)]^T,$$

$$(A.1c) \quad \bar{f}(x) \triangleq [f_1^T(x) f_2^T(x) \cdots f_m^T(x)]^T.$$

Since $\psi(\hat{x}) = 0$ by assumption, $f_k^j(\hat{x}) = 0$ for all $k \in \hat{\mathbf{r}}(\hat{x})$ and all $j \in \mathbf{q}_k(\hat{x})$. Hence,

$$(A.1d) \quad \bar{f}(\hat{x}) = 0.$$

In view of Assumption 3.9(a), we also can assume, without loss of generality, that the Jacobian matrix $\bar{f}_x(\hat{x})$ can be written as $[J_1 \ J_2]$, where J_1 is a nonsingular $\bar{p} \times \bar{p}$ matrix.

Next, we define the $n \times n$ matrix $\hat{M}$ by

$$(A.1e) \quad \hat{M} \triangleq - \begin{bmatrix} J_1 & J_2 \\ 0 & I_{n-\bar{p}} \end{bmatrix}$$

where, for any $j \in \mathbb{N}$, I_j denotes the $j \times j$ identity matrix. Because J_1 is not singular, neither is $\hat{M}$. Now consider the mapping $H: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ defined by

$$(A.1f) \quad H(z, x) \triangleq z - \begin{bmatrix} \bar{f}(x) \\ x^{\bar{p}+1} - \hat{x}^{\bar{p}+1} \\ \vdots \\ x^n - \hat{x}^n \end{bmatrix}.$$

Clearly, (A.1d–e) imply that $H(0, \hat{x}) = 0$ and $DH(0, \hat{x}) = [I_n \ \hat{M}]$. Moreover, because the functions $f_k^j(\cdot)$ are twice continuously differentiable, so is $H(\cdot, \cdot)$. It follows from the implicit function theorem that there exists a $\hat{\rho} > 0$ and a twice continuously differentiable function $g: B(0, \hat{\rho}) \rightarrow B(\hat{x}, \hat{\rho})$ such that

$$(A.1g) \quad g(0) = \hat{x},$$

$$(A.1h) \quad H(z, g(z)) = 0 \quad \text{for all } z \in B(0, \hat{\rho}),$$

and

$$(A.1i) \quad g_z(0) = \hat{M}^{-1}.$$

Let $\tilde{f}^0: B(0, \hat{\rho}) \rightarrow \mathbb{R}, \tilde{f}_k^j: B(0, \hat{\rho}) \rightarrow \mathbb{R}, k \in \mathbf{r}, j \in \mathbf{q}_k$, be defined by

$$(A.1j) \quad \tilde{f}^0(z) \triangleq f^0(g(z)),$$

$$(A.1k) \quad \tilde{f}_k^j(z) \triangleq f_k^j(g(z)).$$

Because the functions $f^0(\cdot), f_k^j(\cdot)$, and $g(\cdot)$ are twice continuously differentiable, it follows directly from the chain rule and from (A.1j–k) that $\tilde{f}^0(\cdot)$ and $\tilde{f}_k^j(\cdot)$ are also twice continuously differentiable.

Next, we prove that these functions satisfy claims (a)–(e) in Proposition 3.10.

LEMMA A.1. $(0, \hat{\nu}_1, \dots, \hat{\nu}_r)$ is a local minimizer for the problem

$$(3.9a) \quad (\tilde{P}_{\hat{x}}) \quad \min_{(z, \mu_1, \dots, \mu_r)} \left\{ \tilde{f}^0(z) \mid \sum_{j \in \mathbf{q}_k} \mu_k^j \tilde{f}_k^j(z) \leq 0, \mu_k \in \Sigma_{q_k}, k \in \mathbf{r}, z \in B(0, \hat{\rho}) \right\},$$

if and only if $(\hat{x}, \hat{\nu}_1, \dots, \hat{\nu}_r)$ is a local minimizer for $(\tilde{P})$.

Proof. We prove the claim by contraposition. The vector $(0, \hat{\nu}_1, \dots, \hat{\nu}_r)$ is not a local minimizer for $(\tilde{P}_{\hat{x}})$ if and only if there exists a sequence $\{(z_i, \nu_{1,i}, \dots, \nu_{r,i})\}_{i=0}^\infty$, converging to $(0, \hat{\nu}_1, \dots, \hat{\nu}_r)$ as $i \rightarrow \infty$ and such that for all $i \in \mathbb{N}$, $\nu_{k,i} \in \Sigma_{q_k}, k \in \mathbf{r}, z_i \in B(0, \hat{\rho})$,

$$(A.2a) \quad \tilde{f}^0(z_i) < \tilde{f}^0(0)$$

and

$$(A.2b) \quad \sum_{j \in \mathbf{q}_k} \nu_{k,i}^j \tilde{f}_k^j(z_i) \leq 0, \quad k \in \mathbf{r}.$$

If for each i we define $x_i \triangleq g(z_i)$, we conclude, using (A.1j–k) and (A.2a–b), that $(0, \hat{\nu}_1, \dots, \hat{\nu}_r)$ is not a local minimizer for $(\tilde{P}_{\hat{x}})$ if and only if there exists a sequence $\{(x_i, \nu_{1,i}, \dots, \nu_{r,i})\}_{i=0}^\infty$, converging to $(\hat{x}, \hat{\nu}_1, \dots, \hat{\nu}_r)$ as $i \rightarrow \infty$ and such that, for all i ,

$$(A.2c) \quad f^0(x_i) = f^0(g(z_i)) = \tilde{f}^0(z_i) < \tilde{f}^0(0) = f^0(g(0)) = f^0(\hat{x})$$

and

$$(A.2d) \quad \sum_{j \in \mathbf{q}_k} \nu_{k,i}^j f_k^j(x_i) = \sum_{j \in \mathbf{q}_k} \nu_{k,i}^j f_k^j(g(z_i)) = \sum_{j \in \mathbf{q}_k} \nu_{k,i}^j \tilde{f}_k^j(z_i) \leq 0.$$

It follows that $(0, \hat{\nu}_1, \dots, \hat{\nu}_r)$ is not a local minimizer for $(\tilde{P}_{\hat{x}})$ if and only if $(\hat{x}, \hat{\nu}_1, \dots, \hat{\nu}_r)$ is not a local minimizer for $(\tilde{P})$, which implies the desired result by contraposition. $\square$

LEMMA A.2. *There exists a one-to-one map $l(\cdot, \cdot)$ taking pairs (k, j) , with $k \in \hat{\mathbf{r}}(\hat{x})$ and $j \in \hat{\mathbf{q}}_k(\hat{x})$, into $\mathbf{n} \triangleq \{1, \dots, n\}$, and such that*

$$(3.9b) \quad \tilde{f}_k^j(z) = z^{l(k,j)}, \quad k \in \hat{\mathbf{r}}(\hat{x}), \quad j \in \hat{\mathbf{q}}_k(\hat{x}),$$

for all $z \in B(0, \hat{\rho})$.

Proof. For each $k \in \hat{\mathbf{r}}(\hat{x})$ and $j \in \hat{\mathbf{q}}_k(\hat{x})$, let

$$(A.3a) \quad l(k, j) \triangleq \sum_{i=1}^{k-1} p_i + j,$$

where we follow the convention that $\sum_{i=1}^0 p_i = 0$. It follows from (A.1a) and (A.3a) that for each $s \leq \bar{p}$ there exist a unique pair (k, j) with $k \in \hat{\mathbf{r}}(\hat{x})$ and $j \in \hat{\mathbf{q}}_k(\hat{x})$ such that $s = l(k, j)$. Since $l(\cdot, \cdot)$ is strictly monotone increasing in both arguments, it is clear that $l(\cdot, \cdot)$ is one-to-one. Close examination of (A.1b–c) and (A.3a) reveals that for each $k \in \hat{\mathbf{r}}(\hat{x})$ and $j \in \hat{\mathbf{q}}_k(\hat{x})$, $l(k, j)$ is the index of the component of $\bar{f}(\cdot)$ corresponding to $f_k^j(\cdot)$ (see (A.1b–c)), that is,

$$(A.3b) \quad \bar{f}^{l(k,j)}(x) = f_x^j(x)$$

for $x \in B(\hat{x}, \hat{\rho})$, $k \in \hat{\mathbf{r}}(\hat{x})$, and $j \in \hat{\mathbf{q}}_k(\hat{x})$.

From (A.1f–h) we have that, for any $z \in B(0, \hat{\rho})$,

$$(A.3c) \quad H(z, g(z)) = z - \begin{pmatrix} \bar{f}(g(z)) \\ g^{\bar{p}+1}(z) - \hat{x}^{\bar{p}+1} \\ \vdots \\ g^n(z) - \hat{x}^n \end{pmatrix} = 0.$$

Hence, from the first $\bar{p}$ components of (A.3c), (A.3b), (A.1j–k), and (A.3a) we obtain that, for any $z \in B(0, \hat{\rho})$,

$$(A.3d) \quad \tilde{f}_k^j(z) = f_k^j(g(z)) = \bar{f}^{l(k,j)}(g(z)) = z^{l(k,j)}, \quad k \in \hat{\mathbf{r}}(\hat{x}), \quad j \in \hat{\mathbf{r}}(\hat{x}),$$

which completes the proof. $\square$

LEMMA A.3. $\partial \tilde{f}^0(0)/\partial z^s \leq 0$ if $s = l(k, j)$ for some $k \in \hat{\mathbf{r}}(\hat{x})$ and some $j \in \hat{\mathbf{q}}_k(\hat{x})$; $\partial \tilde{f}^0(0)/\partial z^s = 0$ otherwise.

Proof. In view of (A.1a) and (A.3a), it is clear that we need to show that $\partial \tilde{f}^0(0)/\partial z^s \leq 0$ for all $s \leq \bar{p}$ and that $\partial \tilde{f}^0(0)/\partial z^s = 0$ for all $s > \bar{p}$.

First, we show that $\partial \tilde{f}^0(0)/\partial z^s = 0$ for all $s > \bar{p}$. To obtain a contradiction, suppose that there exists a $s^* > \bar{p}$ such that $\partial \tilde{f}^0(0)/\partial z^{s^*} \neq 0$. We will construct a sequence $\{(z_i, \mu_{1,i}, \dots, \mu_{r,i})\}$, converging to $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ as $i \rightarrow \infty$ and feasible for $(\tilde{P}_{\hat{x}})$, such that $\tilde{f}^0(z_i) < \tilde{f}^0(0)$ for all $i \geq i_0$. This implies that $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is not a local minimizer of $(\tilde{P}_{\hat{x}})$, which is a contradiction in view of Lemma A.1.

Let $h \in \mathbb{R}^n$ be such that $h^s = 0$ for all $s \leq \bar{p}$, and $\langle \nabla \tilde{f}^0(\hat{z}), h \rangle = -\varepsilon < 0$. For each i , let $z_i \triangleq h/i$. Clearly, $z_i \rightarrow 0$ as $i \rightarrow \infty$. Let i_0 be such that $z_i \in B(0, \hat{\rho})$ for all $i \geq i_0$. For all $i \geq i_0$, let $x_i = g(z_i)$, with $g(\cdot)$ as in (A.1g). To construct the $\mu_{k,i}$'s, we consider two cases separately: $k \in \hat{\mathbf{r}}^c(\hat{x})$ and $k \in \hat{\mathbf{r}}(\hat{x})$. For each $k \in \hat{\mathbf{r}}^c(\hat{x})$,

$$(A.4a) \quad \min_{j \in \mathbf{q}_k} \tilde{f}_k^j(0) = \min_{j \in \mathbf{q}_k} f_k^j(\hat{x}) < 0.$$

Hence, using (A.1j–k), it follows from Lemma 3.4 that there exist an $i_1 \geq i_0$ and sequences $\{\mu_{k,i}\}_{i=i_1}^\infty \subset \Sigma_{q_k}$, $k \in \hat{\mathbf{r}}^c(\hat{x})$, such that $\mu_{k,i} \rightarrow \hat{\mu}_k$ as $i \rightarrow \infty$, and all $i \geq i_1$,

$$(A.4b) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j \tilde{f}_k^j(z_i) = \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j f_k^j(x_i) \leq 0, \quad k \in \hat{\mathbf{r}}^c(\hat{x}).$$

Next, for each $k \in \hat{\mathbf{r}}(\hat{x})$, let $\mu_{k,i} \triangleq \hat{\mu}_k$, $k \in \hat{\mathbf{r}}(\hat{x})$, for all $i \in \mathbb{N}$. Using Corollary 3.7, Lemma A.2, the definition of $\bar{p}$, and the fact that $h^s = 0$ if $s \leq \bar{p}$, we get that, for each $i \in \mathbb{N}$ and $k \in \hat{\mathbf{r}}(\hat{x})$,

$$(A.4c) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j \tilde{f}_k^j(z_i) = \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \hat{\mu}_k^j \tilde{f}_k^j(z_i) = \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \hat{\mu}_k^j h^{l(k,j)}/i = 0, \quad k \in \hat{\mathbf{r}}(\hat{x}),$$

since $l(k,j) \leq \bar{p}$, and therefore $h^{l(k,j)} = 0$ for all $k \in \hat{\mathbf{r}}(\hat{x})$ and all $j \in \hat{\mathbf{q}}_k(\hat{x})$. Hence, the sequence $\{(z_i, \mu_{1,i}, \dots, \mu_{r,i})\}_{i=i_1}^\infty$ is feasible for $(P_{\hat{x}})$. Furthermore, there exists an $i_2 \geq i_1$ such that

$$(A.4d) \quad \tilde{f}^0(z_i) - \tilde{f}^0(0) = \langle \nabla \tilde{f}^0(0), z_i \rangle + o(\|z_i\|) = -\varepsilon/i + o(i) < 0$$

for all $i \geq i_2$. This implies that $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is not a local minimizer for $(\tilde{P}_{\hat{x}})$, which contradicts Lemma A.1. Hence, we have established that $\partial \tilde{f}^0(0)/\partial z^s = 0$ for all $s > \bar{p}$.

Resorting again to a proof by contradiction, we show next that $\partial \tilde{f}^0(0)/\partial z^s \leq 0$ for all $s \leq \bar{p}$. Thus, suppose that there exists an $s^* \leq \bar{p}$ such that $\partial \tilde{f}^0(0)/\partial z^{s^*} > 0$. Since $s^* \leq \bar{p}$, and $l(\cdot, \cdot)$ is one-to-one, there exist a unique $k^* \in \hat{\mathbf{r}}(\hat{x})$ and a unique $j^* \in \hat{\mathbf{q}}_{k^*}(\hat{x})$, such that $l(k^*, j^*) = s^*$. For each i , let

$$(A.4e) \quad z_i^{l(k^*, j^*)} \triangleq -1/i,$$

$$(A.4f) \quad z_i^{l(k,j)} \triangleq 0 \quad \text{if } (k,j) \neq (k^*, j^*).$$

For each i such that $z_i \in B(0, \hat{\rho})$, let $x_i \triangleq g(z_i)$. Next, let $\mu_{k,i} \triangleq \hat{\mu}_k$, $k \in \hat{\mathbf{r}}(\hat{x})$. Then, using (A.4f) and Corollary 3.7, we get that

$$(A.4g) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j \tilde{f}_k^j(z_i) = \sum_{j \in \mathbf{q}_k(\hat{x})} \hat{\mu}_k^j z_i^{l(k,j)} = 0,$$

for each $k \in \hat{\mathbf{r}}(\hat{x}) \setminus \{k^*\}$, and in view of (A.1k),

$$(A.4h) \quad \sum_{j \in \mathbf{q}_{k^*}} \mu_{k^*,i}^j f_{k^*}^j(x_i) = \sum_{j \in \mathbf{q}_{k^*}} \mu_{k^*,i}^j \tilde{f}_{k^*}^j(z_i) = \sum_{j \in \mathbf{q}_{k^*}(\hat{x})} \hat{\mu}_{k^*}^j z_i^{l(k^*, j)} = -\mu_{k^*}^{j^*}/i \leq 0.$$

It follows from Lemma 3.4 that there exist an i_0 and sequences $\{\mu_{k,i}\}_{i=i_0}^\infty \subset \Sigma_{q_k}$, $k \in \hat{\mathbf{r}}^c(\hat{x})$, such that $\mu_{k,i} \rightarrow \hat{\mu}_k$ as $i \rightarrow \infty$, and for all $i \geq i_0$,

$$(A.4i) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j \tilde{f}_k^j(z_i) \leq 0, \quad k \in \hat{\mathbf{r}}^c(\hat{x}),$$

which, together with (A.4g–h), implies that $\{(z_i, \mu_{1,i}, \dots, \mu_{r,i})\}_{i=i_0}^\infty$ is feasible for $\tilde{P}'_{\hat{x}}$.

Clearly, $(z_i, \mu_{1,i}, \dots, \mu_{r,i}) \rightarrow (0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ as $i \rightarrow \infty$. Moreover, in view of (A.4e–f) and the fact that $\partial \tilde{f}^0(0)/\partial z^{j*} > 0$, we have that

$$(A.4j) \quad \langle \nabla \tilde{f}^0(0), z_i \rangle = \sum_{j \in \mathbf{n}} \frac{\partial \tilde{f}^0(0)}{\partial z^j} z_i^j = -\frac{1}{i} \frac{\partial \tilde{f}^0(0)}{\partial z^{j*}} < 0$$

for all i . It follows that there exists $i_1 \geq i_0$ such that $\tilde{f}^0(z_i) < \tilde{f}^0(0)$ for all $i \geq i_1$. Hence, $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is not a local minimizer for $(\tilde{P}'_{\hat{x}})$, which contradicts Lemma A.1. $\square$

LEMMA A.4. *For each $k \in \hat{\mathbf{r}}(\hat{x})$ and each $j \in \hat{\mathbf{q}}_k(\hat{x})$,*

$$\hat{\mu}_k^j \sum_{s \in \hat{\mathbf{q}}_k(\hat{x})} \left| \frac{\partial \tilde{f}^0}{\partial z^{l(k,s)}}(0) \right| = -\frac{\partial \tilde{f}^0}{\partial z^{l(k,j)}}(0).$$

Proof. For each $k^* \in \hat{\mathbf{r}}(\hat{x})$, let

$$(A.4k) \quad S_{k^*} \triangleq \sum_{s \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \left| \frac{\partial \tilde{f}^0}{\partial z^{l(k^*,s)}}(0) \right|.$$

If $S_{k^*} = 0$, then the claim is trivially true since in this case we must have $\partial \tilde{f}^0(0)/\partial z^{l(k^*,j)} = 0$ for all $j \in \hat{\mathbf{q}}_{k^*}(\hat{x})$, which clearly implies the desired result.

Hence, suppose that $k^* \in \hat{\mathbf{r}}(\hat{x})$ and $S_{k^*} \neq 0$. Let $\hat{\lambda}_{k^*} \in \mathbb{R}^{q_{k^*}}$ be defined by

$$(A.4l) \quad \hat{\lambda}_{k^*}^j \triangleq -\frac{1}{S_{k^*}} \frac{\partial \tilde{f}^0(0)}{\partial z^{l(k^*,j)}}, \quad j \in \hat{\mathbf{q}}_{k^*}(\hat{x}),$$

$$(A.4m) \quad \hat{\lambda}_{k^*}^j \triangleq 0, \quad j \in \hat{\mathbf{q}}_{k^*}^c(\hat{x}).$$

It follows from Lemma A.3 that $\partial \tilde{f}^0(0)/\partial z^{l(k^*,j)} \leq 0$ for all $j \in \hat{\mathbf{q}}_{k^*}(\hat{x})$, and hence (A.4k–m) imply that $\lambda_{k^*} \in \Sigma_{q_{k^*}}$.

Next, we will show that $\hat{\lambda}_{k^*} = \hat{\mu}_{k^*}$. To obtain a contradiction, suppose that $\hat{\lambda}_{k^*} \neq \hat{\mu}_{k^*}$. In this case, there exists $\nu \in \mathbb{R}^{q_{k^*}}$, $\nu \neq 0$, and $\varepsilon > 0$ such that $\langle \hat{\mu}_{k^*}, \nu \rangle \leq 0$ and $\langle \hat{\lambda}_{k^*}, \nu \rangle = \varepsilon > 0$.

First, for any $k \in \hat{\mathbf{r}}(\hat{x})$ and $i \in \mathbb{N}$, we set $\mu_{k,i} \triangleq \hat{\mu}_k$ and define $\{z_i\}_{i=i_0}^\infty$ by

$$(A.4n) \quad z_i^{l(k^*,j)} \triangleq \nu^j / i, \quad j \in \hat{\mathbf{q}}_{k^*}(\hat{x}),$$

$$(A.4o) \quad z_i^s \triangleq 0, \quad s \notin \bigcup_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \{l(k^*,j)\}.$$

Since $k^* \in \hat{\mathbf{r}}(\hat{x})$, it follows from Corollary 3.7, (3.9b), and (A.4n–o) that, for all i ,

$$(A.4p) \quad \begin{aligned} \sum_{j \in \mathbf{q}_{k^*}} \mu_{k^*,i}^j \tilde{f}_{k^*}^j(z_i) &= \sum_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \hat{\mu}_{k^*}^j \tilde{f}_{k^*}^j(z_i) \\ &= \sum_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \hat{\mu}_{k^*}^j z_i^{l(k^*,j)} = \sum_{j \in \hat{\mathbf{q}}_{k^*}(\hat{x})} \hat{\mu}_{k^*}^j \frac{\nu^j}{i} = \frac{1}{i} \langle \hat{\mu}_{k^*}, \nu \rangle \leq 0. \end{aligned}$$

Moreover, again using Corollary 3.7, (3.9b), and (A.4o), it follows that, for all $k \in \hat{\mathbf{r}}(\hat{x}) \setminus \{k^*\}$,

$$(A.4q) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j \tilde{f}_k^j(z_i) = \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \hat{\mu}_k^j \tilde{f}_k^j(z_i) = \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \hat{\mu}_k^j z_i^{l(k,j)} = 0.$$

Next we turn to the indices $k \in \hat{\mathbf{r}}^c(\hat{x})$. It follows from Lemma 3.3 that there exist an i_0 and sequences $\{\mu_{k,i}\}_{i=i_0}^\infty$, $k \in \hat{\mathbf{r}}^c(\hat{x})$, such that $\mu_{k,i} \in \Sigma_{q_k}$, $\mu_{k,i} \rightarrow \hat{\mu}_k$, $k \in \hat{\mathbf{r}}^c(\hat{x})$, and for all $i \geq i_0$,

$$(A.4r) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j \tilde{f}_k^{j,j}(z_i) \leq 0, \quad k \in \hat{\mathbf{r}}^c(\hat{x}).$$

It follows from (A.4p–r) that $\{(z_i, \mu_i^1, \dots)\}_{i=i_0}^\infty$ is feasible for $(\tilde{\mathbf{P}}'_\hat{x})$. Moreover, using (A.4n–o), (A.4l), and the fact that $\langle \hat{\lambda}_{k^*}, \nu \rangle = \varepsilon$, there exists an $i_1 \geq i_0$ such that

$$(A.4s) \quad \begin{aligned} \tilde{f}^0(z_i) - \tilde{f}^0(0) &= \langle \nabla \tilde{f}^0(0), z_i - 0 \rangle + o(\|z_i - 0\|) = \frac{1}{i} \sum_{j \in \mathbf{q}_{k^*}} \frac{\partial \tilde{f}^0(0)}{\partial z^{l(k^*,j)}} \nu^j + o(1/i) \\ &= -\frac{S_{k^*}}{i} \sum_{j \in \mathbf{q}_{k^*}} \hat{\lambda}_{k^*}^j \nu^j + o(1/i) = -\frac{\varepsilon S_{k^*}}{i} + o(1/i) \leq 0 \end{aligned}$$

for all $i \geq i_1$. This implies that $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is not a local minimizer for $(\tilde{\mathbf{P}}'_\hat{x})$, which contradicts Lemma A.1. Hence, we must have that $\hat{\lambda}_{k^*} = \hat{\mu}_{k^*}$. In view of (A.4k–l), this implies the desired result. $\square$

We will need the following lemma to prove the last result in this appendix.

LEMMA A.5. *Let $\nu_k \in \Sigma_{q_k}$, $k \in \mathbf{r}$, be such that*

$$(A.5a) \quad \nu_k = \hat{\mu}_k, \quad k \in \hat{\mathbf{r}}(\hat{x}),$$

and

$$(A.5b) \quad \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \nu_k^j = 1, \quad k \in \hat{\mathbf{r}}^c(\hat{x}).$$

Then $(0, \nu_1, \dots, \nu_r)$ is a local minimizer for $(\tilde{\mathbf{P}}'_\hat{x})$.

Proof. First, since $(\hat{x}, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}})$, we have, by Lemma A.1, that $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{\mathbf{P}}'_\hat{x})$. Next, to obtain a contradiction, suppose that $(0, \nu_1, \dots, \nu_r)$ is not a local minimizer for $(\tilde{\mathbf{P}}'_\hat{x})$. Clearly, by the choice of $\nu_k, k \in \mathbf{r}$, we have that $(0, \nu_1, \dots, \nu_r)$ is feasible for $(\tilde{\mathbf{P}}'_\hat{x})$, and hence there must exist sequences $\{z_i\}_{i=0}^\infty$ and $\{\nu_{k,i}\}_{i=0}^\infty$, $k \in \hat{\mathbf{r}}(\hat{x})$, such that $x_i \rightarrow \hat{x}$ and $\nu_{k,i} \rightarrow \nu_k = \hat{\mu}_k, k \in \hat{\mathbf{r}}(\hat{x})$, as $i \rightarrow \infty$, and for all i ,

$$(A.5c) \quad f^0(z_i) < f^0(0)$$

and

$$(A.5d) \quad \sum_{j \in \mathbf{q}_k} \nu_{k,i}^j \tilde{f}_k^j(z_i) \leq 0, \quad k \in \hat{\mathbf{r}}(\hat{x}).$$

Now, because $\psi(\hat{x}) = 0$, for each $k \in \hat{\mathbf{r}}^c(\hat{x})$, $\psi_k(\hat{x}) < 0$. It follows that, for each $k \in \hat{\mathbf{r}}^c(\hat{x})$,

$$(A.5e) \quad \min_{j \in \mathbf{q}_k} \tilde{f}_k^j(0) = \min_{j \in \mathbf{q}_k} f_k^j(\hat{x}) = \psi_k(\hat{x}) < 0.$$

Since $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is feasible for $(\tilde{P})$, we have

$$(A.5f) \quad \sum_{j \in \mathbf{q}_k} \hat{\mu}_k^j \tilde{f}_k^j(0) = \sum_{j \in \mathbf{q}_k} \hat{\mu}_k^j f_k^j(\hat{x}) \leq 0, \quad k \in \hat{\mathbf{r}}^c(\hat{x}).$$

Using Lemma 3.4, (A.5e) and (A.5f) imply that there exist an i_0 and sequences $\{\mu_{k,i}\}_{i=i_0}^\infty$, $k \in \hat{\mathbf{r}}^c(\hat{x})$, such that $\mu_{k,i} \rightarrow \hat{\mu}_k$, $k \in \hat{\mathbf{r}}^c(\hat{x})$, as $i \rightarrow \infty$, and, for all $i \in \mathbb{N}$,

$$(A.5g) \quad \sum_{j \in \mathbf{q}_k} \mu_{k,i}^j \tilde{f}_k^j(z_i) \leq 0, \quad k \in \hat{\mathbf{r}}^c(\hat{x}).$$

This, together with (A.5c) and (A.5d), implies that $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is not a local minimizer for $(\tilde{P}')$, which is a contradiction. $\square$

LEMMA A.6. *For each $k \in \hat{\mathbf{r}}(\hat{x})$, there exists a $j_k \in \hat{\mathbf{q}}_k(\hat{x})$ such that $\hat{\mu}^{k,j_k} > 0$ and*

$$\frac{\partial \tilde{f}^0}{\partial z^{l(k,j_k)}}(0) < 0.$$

Proof. We will show that for any $k \in \hat{\mathbf{r}}(\hat{x})$ and any $j \in \hat{\mathbf{q}}_k(\hat{x})$, $\partial \tilde{f}^0(0)/\partial z^{l(k,j)} = 0$ if and only if $\mu_k^j = 0$. This, together with Corollary 3.7 and Lemma A.3, implies the desired result.

First, since by Lemma A.1 $(0, \hat{\mu}_1, \dots, \hat{\mu}_r)$ is a local minimizer for $(\tilde{P}'_{\hat{x}})$, so is $(0, \nu_1, \dots, \nu_r)$, where the ν_k 's are defined as in Lemma A.5. From (A.5b) we have that

$$(A.6a) \quad \nu_k^j = 0, \quad j \in \hat{\mathbf{q}}_k^c(\hat{x}), \quad k \in \mathbf{r}.$$

Because $\psi(\hat{x}) = 0$, $\psi_k(\hat{x}) < 0$ for each $k \in \hat{\mathbf{r}}^c(\hat{x})$. Therefore, it follows from (A.6a) that, for all $k \in \hat{\mathbf{r}}^c(\hat{x})$,

$$(A.6b) \quad \begin{aligned} \sum_{j \in \mathbf{q}_k} \nu_k^j \tilde{f}_k^j(\hat{z}) &= \sum_{j \in \mathbf{q}_k} \nu_k^j f_k^j(\hat{x}) = \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \nu_k^j f_k^j(\hat{x}) \\ &= \psi_k(\hat{x}) \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \nu_k^j = \psi_k(\hat{x}) < 0. \end{aligned}$$

Next, from the Fritz-John first-order necessary conditions of optimality for $(\tilde{P}'_{\hat{x}})$ at $(0, \nu_1, \dots, \nu_r)$, it follows that there exist multipliers $\hat{\zeta}^j \in [0, 1]$, $j \in \{0, 1, \dots, r\}$, such that

$$(A.6c) \quad \hat{\zeta}^0 \nabla \tilde{f}^0(0) + \sum_{k \in \mathbf{r}} \hat{\zeta}^k \sum_{j \in \mathbf{q}_k} \nu_k^j \nabla f_k^j(0) = 0$$

and

$$(A.6d) \quad \hat{\zeta}^k \sum_{j \in \mathbf{q}_k} \nu_k^j \tilde{f}_k^j(0) = 0, \quad k \in \mathbf{r}.$$

It follows from (A.6b) and (A.6d) that $\hat{\zeta}^k = 0$ for all $k \in \hat{\mathbf{r}}^c(\hat{x})$. Hence, in view of (A.6a) and the fact that $\nu_k = \hat{\mu}_k$ for $k \in \hat{\mathbf{r}}(\hat{x})$ by construction, (A.6c) implies that

$$(A.6e) \quad \hat{\zeta}^0 \nabla \tilde{f}^0(0) + \sum_{k \in \hat{\mathbf{r}}(\hat{x})} \hat{\zeta}^k \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \hat{\mu}_k^j \nabla f_k^j(0) = 0.$$

On the one hand, it follows from (A.6e) and from Assumption 3.9(a) that we must have $\hat{\zeta}^0 > 0$. Hence, we can define $\tilde{\zeta}^k \triangleq \hat{\zeta}^k / \hat{\zeta}^0$ and rewrite (A.6e) as

$$(A.6f) \quad \nabla \tilde{f}^0(0) + \sum_{k \in \hat{\mathbf{r}}(\hat{x})} \tilde{\zeta}^k \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \hat{\mu}_k^j \nabla \tilde{f}_k^j(0) = 0.$$

On the other hand, (A.6f) and Assumption 3.9(b) imply that $\tilde{\zeta}^k > 0$ for all $k \in \hat{\mathbf{r}}(\hat{x})$. Recalling that by (3.8b), $\tilde{f}_k^j(0) = z^{l(k,j)}$ for each $k \in \hat{\mathbf{r}}(\hat{x})$ and each $j \in \hat{\mathbf{q}}_k(\hat{x})$, we obtain from (A.6f) that, for each $k \in \hat{\mathbf{r}}(\hat{x})$ and $j \in \hat{\mathbf{q}}_k(\hat{x})$,

$$(A.6g) \quad \frac{\partial \tilde{f}^0}{\partial z^{l(k,j)}}(0) + \tilde{\zeta}^k \sum_{j \in \hat{\mathbf{q}}_k(\hat{x})} \hat{\mu}_k^{l(k,j)} = 0.$$

Since $\tilde{\zeta}^k > 0$ for all $k \in \hat{\mathbf{r}}(\hat{x})$, (A.6g) implies that, for all $k \in \hat{\mathbf{r}}(\hat{x})$ and all $j \in \hat{\mathbf{q}}_k(\hat{x})$, $\hat{\mu}_k^j = 0$ if and only if

$$(A.6h) \quad \frac{\partial \tilde{f}^0}{\partial z^{l(k,j)}}(0) = 0,$$

which implies the desired result. $\square$

Proposition 3.10 follows directly from Lemmas A.1–A.4 and A.6.

REFERENCES

- [BaL.1] J. W. BANDLER, P. C. LIU, AND H. TROMP, *A nonlinear programming approach to optimal design centering, tolerancing, and tuning*, IEEE Trans. Circuits Systems, CAS-23 (1976), pp. 155–165.
- [BrM.1] J. BRACKEN AND J. T. MCGILL, *Defense applications of mathematical programs with optimization problems in the constraints*, Oper. Res., 22 (1974), pp. 1086–1096.
- [ChD.1] C. K. CHENG, X. DENG, Y. Z. LIAO, AND S. Z. YAO, *Symbolic layout compaction under conditional design rules*, IEEE Trans. Comput. Aided Design, 11 (1992), pp. 475–486.
- [GiJ.1] E. G. GILBERT AND D. W. JOHNSON, *Distance functions and their application to robot path planning in the presence of obstacles*, IEEE J. Robotics Automat., RA-1 (1985), pp. 21–30.
- [GiM.1] P. E. GILL, W. MURRAY, M. A. SAUNDERS, AND M. H. WRIGHT, *User's Guide for NPSOL: A Fortran Package for Nonlinear Programming*, Technical Report 86-2, Dept. of Operations Research, Stanford University, Stanford, CA, 1986.
- [GrS.1] I. E. GROSSMANN AND R. W. SARGENT, *Optimal design of chemical plants with uncertain parameters*, AIChE J., 24 (1978), pp. 1–7.
- [HaG.1] K. P. HALEMANE AND I. E. GROSSMAN, *Optimal process design under uncertainty*, AIChE J., 29 (1983), pp. 425–433.
- [HeP.1] L. HE AND E. POLAK, *Effective diagonalization strategies for the solution of a class of optimal design problems*, IEEE Trans. Automat. Control, 35 (1990), pp. 258–267.
- [Hoc.1] D. HOCHBAUM, *Complexity and algorithms for logical constraints with applications to VLSI layout, compaction and clocking*, Studies in Locational Analysis, ISOLD VI Proceedings, 1993, pp. 159–164.
- [IeP.1] M. G. IERAPETRITOU AND E. N. PISTIKOPOULOS, *Simultaneous incorporation of flexibility and economic risk in operational planning under uncertainty*, Comput. Chem. Engrg., 18 (1994), pp. 163–189.
- [KiP.1] C. KIRJNER-NETO AND E. POLAK, *On the use of consistent approximations for the optimal design of beams*, SIAM J. Control Optim., 34 (1996), pp. 1891–1913.
- [LiC.1] P. C. LIU, V. W. CHUNG, AND K. C. LI, *Circuit design with post-fabrication tuning*, in Proc. 35th Midwest Symposium on Circuits and Systems, Washington, DC, IEEE, NY, 1992, pp. 344–347.
- [MaP.1] D. Q. MAYNE AND E. POLAK, *Algorithms for optimization problems with exclusion constraints*, J. Optim. Theory Appl., 51 (1986), pp. 453–474.

- [MaP.2] D. Q. MAYNE AND E. POLAK, *Outer approximations algorithm for nondifferentiable optimization problems*, J. Optim. Theory Appl., 42 (1984), pp. 19–30.
- [MoG.1] J. J. MORE, B. S. GARBOW, AND K. E. HILLSTROM, *Testing unconstrained optimization software*, ACM Trans. Math. Software, 7 (1981), pp. 17–41.
- [Mul.1] G. MULLER, *On computer-aided tuning of microwave filters*, in IEEE Proc. International Symposium on Circuits and Systems, Munich, IEEE Computer Society Press, Los Alamitos, CA, 1976, pp. 722–725.
- [OsV.1] G. M. OSTROVSKY, Y. M. VOLIN, E. I. BARIT, AND M. M. SENYAVIN, *Flexibility analysis and optimization of chemical plants with uncertain parameters*, Comput. Chem. Engrg., 18 (1994), pp. 755–767.
- [PoS.1] E. POLAK AND A. SANGIOVANNI VINCENTELLI, *Theoretical and computational aspects of the optimal design centering, tolerancing and tuning problem*, IEEE Trans. Circuits Systems, CAS-26 (1979), pp. 795–813.
- [Pol.1] E. POLAK, *An implementable algorithm for the design centering, tolerancing and tuning problem*, J. Optim. Theory Appl., 35 (1981), pp. 45–67.
- [Pol.2] E. POLAK, *On the use of consistent approximations in the solution of semi-infinite optimization and optimal control problems*, Math. Programming Ser. B, 62 (1993), pp. 385–414.
- [Pol.3] E. POLAK, *Optimization: Algorithms and Consistent Approximations*, Springer-Verlag, New York, 1997.
- [ReS.1] J. A. REEDS AND L. A. SHEPP, *Optimal paths for a car that goes both forwards and backwards*, Pacific J. Math., 145 (1990), pp. 75–90.
- [Tit.1] A. TITS, *On the optimal design centering, tolerancing and tuning problem*, J. Optim. Theory Appl., 45 (1985), pp. 487–494.

CURVILINEAR STABILIZATION TECHNIQUES FOR TRUNCATED NEWTON METHODS IN LARGE SCALE UNCONSTRAINED OPTIMIZATION*

STEFANO LUCIDI[†], FRANCESCO ROCHETICH[†], AND MASSIMO ROMA[†]

Abstract. The aim of this paper is to define a new class of minimization algorithms for solving large scale unconstrained problems. In particular we describe a stabilization framework, based on a curvilinear linesearch, which uses a combination of a Newton-type direction and a negative curvature direction. The motivation for using negative curvature direction is that of taking into account local nonconvexity of the objective function. On the basis of this framework, we propose an algorithm which uses the Lanczos method for determining at each iteration both a Newton-type direction and an effective negative curvature direction. The results of extensive numerical testing are reported together with a comparison with the LANCELOT package. These results show that the algorithm is very competitive, which seems to indicate that the proposed approach is promising.

Key words. large scale unconstrained optimization, Newton-type method, negative curvature direction, curvilinear linesearch, Lanczos method

AMS subject classifications. 90C30, 65K05

PII. S1052623495295250

1. Introduction. In this work, we deal with the definition of new efficient unconstrained minimization algorithms for solving large scale problems. Namely, we consider the minimization problem

$$(1.1) \quad \min_{x \in \mathbb{R}^n} f(x),$$

where f is a real valued function on $\mathbb{R}^n$ and n is large. We assume throughout that both the gradient $g(x) = \nabla f(x)$ and the Hessian matrix $H(x) = \nabla^2 f(x)$ of f exist and are continuous.

Many algorithms have been proposed for solving such problems; in particular, we focus our attention on truncated Newton-type algorithms since they retain the good Newton-like convergence rate properties.

Most of the Newton-type algorithms for large scale minimization problems perform quite well on test problems which present small regions where the objective function is nonconvex, but they have trouble in minimizing test functions where these regions are large. This behavior can be clearly evidenced by running these algorithms on the CUTE collection [2], which includes a variety of functions with large nonconvexity regions. As far as the authors are aware, the only Newton-type algorithm which performs sufficiently well on all the CUTE test problems is the truncated Newton algorithm implemented in the LANCELOT software package [3]. In our opinion this is due to the fact that, unlike the other ones, this algorithm is able to take the local nonconvexity of the objective function into account. The LANCELOT algorithm is a trust region method, and at each iteration k , its idea is to compute a step using a truncated conjugate gradient method to approximately minimize the quadratic model of

*Received by the editors November 28, 1995; accepted for publication (in revised form) September 10, 1997; published electronically August 26, 1998. This work was partially supported by Agenzia Spaziale Italiana, Roma, Italy.

<http://www.siam.org/journals/siopt/8-4/29525.html>

[†]Dipartimento di Informatica e Sistemistica, Università di Roma “La Sapienza,” Via Buonarroti 12, 00185 Roma, Italy (lucidi@dis.uniroma1.it, roma@dis.uniroma1.it).

the objective function within a trust region. However, whenever the conjugate gradient method detects a nonascent negative curvature direction (i.e., a direction d_k such that $g(x_k)^T d_k \leq 0$ and $d_k^T H(x_k) d_k < 0$), the minimization of the model is stopped and the new point is obtained by performing a significant movement along this direction. Since the direction d_k is one in which the function decreases and its directional derivative also is decreasing (at least locally), this movement can contribute significantly to hasten the search for a region in which the objective function is convex so that the properties of a Newton-type method may apply.

The previous remarks seem to indicate that the use of the negative curvature directions could be very beneficial in solving large scale minimization problems. This encourages further investigation in defining new algorithms based on the attempt to exploit more deeply the information about the curvature of f , which can be obtained by means of negative curvature directions. In fact, although the strategy of the LANCELOT algorithm to compute a negative curvature direction is very simple and efficient, some of its aspects can be improved. In particular, even if we are in a region where the Hessian is not positive semidefinite, the conjugate gradient scheme of the LANCELOT algorithm can be truncated before finding a negative curvature direction. Furthermore, even if such a direction is detected, it could convey poor information about the negative curvature of f , and hence, it could be unable to influence the behavior of the algorithm significantly. Therefore, it could be worthwhile, from the computational point of view, to define new algorithms that, without an excessive computational burden, try to use negative curvature directions more effective than the simple ones produced in the LANCELOT algorithm.

Negative curvature directions which can be considered the best from the theoretical point of view are those directions that guarantee the skill of the minimization algorithm in escaping regions where the objective function is strictly nonconvex. Such directions are the basis for defining unconstrained minimization algorithms which converge toward stationary points where the Hessian matrix is positive semidefinite. This second order convergence can be ensured by exploiting, to a great extent, the negative curvature information of the objective function contained in the second order derivatives. Such information is obtained by requiring that the negative curvature direction has some resemblance to an eigenvector of the Hessian associated with its minimum eigenvalue. More in particular, it is usually needed to compute a direction of negative curvature d_k such that (see, for example, [15, 16, 17, 24])

$$(1.2) \quad d_k^T H(x_k) d_k \rightarrow 0 \quad \text{implies} \quad \min[0, \lambda_{\min}(H(x_k))] \rightarrow 0,$$

where $\lambda_{\min}(H(x_k))$ is the minimum eigenvalue of the Hessian matrix.

Algorithms converging toward stationary points where the Hessian matrix is positive semidefinite have been defined both in a trust region approach (see, for example, [9, 24]) and in a linesearch approach (see, for example, [15, 16, 17]). In particular, the second approach originated from [15] and [17], which propose a minimization method based on a monotone curvilinear linesearch of the following form:

$$(1.3) \quad x_{k+1} = x_k + \alpha_k^2 s_k + \alpha_k d_k,$$

where α_k is a step size computed by an Armijo-type technique, s_k is a Newton-type direction which ensures global convergence and a superlinear rate of convergence, and d_k is a negative curvature direction which guarantees convergence toward points where the Hessian matrix is positive semidefinite.

In [8] the approach proposed in [15, 17] has been embedded in a nonmonotone stabilization strategy by defining a new general algorithmic model of the type of [13]. The motivation of this extension arose from the fact that substantial computational benefits can be obtained by abandoning the enforcement of generating monotonically decreasing objective function values. These benefits are particularly noticeable in solving minimization problems with highly nonlinear or ill-conditioned objective functions. The effectiveness of using a nonmonotone strategy was shown both in the linesearch approach (see, for example, [11, 13, 27]) and in the trust region approach (see, for example, [7, 28, 29]). In [8] a sample algorithm, based on the nonmonotone curvilinear algorithmic model proposed in this paper, has been implemented and shown to outperform other standard Newton-type algorithms, in particular the LANCELOT algorithm, on a large set of small and medium dimension test problems. These numerical results seem to suggest that the combined use of a nonmonotone stabilization technique and a good negative curvature direction can be the basis for defining new efficient unconstrained minimization algorithms.

The aim of this work is to extend the approach proposed in [8] to the case of large scale problems. The main difficulty in tackling such minimization problems is the fact that the use of condition (1.2) as a criterion for selecting a good negative curvature direction is not practical. In fact, the computation of a direction which satisfies (1.2) requires heavy and expensive matrix manipulations, which limits its applicability when the dimension of the problems becomes large.

In section 2 of this paper, as a first step to overcome this difficulty, we prove that the conditions required on the directions of negative curvature to guarantee the convergence toward stationary points where the Hessian matrix is positive semidefinite can be weakened. In fact, we show that the convergence properties of the general stabilization framework of [8] (hence of the algorithms proposed in [15, 17]) still hold under weaker assumptions on the negative curvature directions used. In particular, these new conditions imply that it is sufficient to compute directions of negative curvature d_k which satisfy

$$(1.4) \quad \begin{cases} d_k^T H(x_k) d_k \rightarrow 0 \\ g(x_k) \rightarrow 0 \end{cases} \quad \text{implies} \quad \min [0, \lambda_{\min}(H(x_k))] \rightarrow 0.$$

This result, besides being a step toward the definition of large scale minimization algorithms which converge toward stationary points where the Hessian matrix is positive semidefinite, gives useful hints on how to compute efficiently a good direction of negative curvature. In fact, roughly speaking, condition (1.4) indicates that an algorithm based on the curvilinear search (1.3) is able to leave a region where the objective function is nonconvex and its gradient is not too small by exploiting only the properties of the direction s_k . Therefore, an expensive computation of a negative curvature direction d_k which satisfies a condition similar to (1.2) is not necessary and justified when the gradient is sufficiently large. On the other hand, the contribution of a direction d_k which has a quite strict connection with an eigenvector of the Hessian corresponding to the most negative eigenvalue is essential to go away from a region of nonconvexity of the objective function in which the gradient is small.

In section 3 of this paper, drawing our inspiration from the previous result, we define a new minimization algorithm for large scale unconstrained problems which, at each iteration, computes both a truncated Newton direction and a negative curvature direction using the Lanczos method to approximately solve the Newton equation.

The choice of using the Lanczos algorithm as the iterative method for computing the search directions is due to the fact that some of its properties are very suitable for the approach described in this work. First of all, it does not break down if the Hessian matrix is indefinite. This implies that a good approximation of the Newton direction can be always calculated and, furthermore, that the process of gaining information on the negative curvature of the objective function is not stopped if the algorithm meets a negative curvature direction. Another interesting feature of the Lanczos algorithm which will be exploited in this paper is the possibility of obtaining good estimates of the smallest eigenvalue of the Hessian matrix and of one of the associated eigenvectors. In fact, if enough storage room is available to store a sequence of vectors produced by the Lanczos algorithm and if suitable assumptions are satisfied, then a negative curvature direction satisfying the new condition described in this work can be determined and, hence, the convergence of the algorithm toward stationary points where the Hessian matrix is positive semidefinite can be ensured.

In practice, in the case of large scale problems, this storage requirement cannot be satisfied and only a small number of vectors produced by the algorithm can be stored. However, also in this case, the theory and the computational experiences guarantee that the Lanczos algorithm can provide efficiently a negative curvature direction that, even if it does not satisfy condition (1.4), can be considered a good approximation of an eigenvector of the Hessian corresponding to the most negative eigenvalue. Therefore, we can conclude that the Lanczos algorithm allows us to compute a negative curvature direction which is a compromise between

- a negative curvature direction which satisfies condition (1.4) but which could be too expensive to compute when the gradient of f becomes small,
- a negative curvature direction (like the one used by the LANCELOT algorithm) which is easy to compute but which could have no resemblance to any eigenvector of the Hessian corresponding to the most negative eigenvalue.

Finally, in section 4, we report the results of a large computational experimentation. One of the purposes of these numerical experiments has been to investigate the effect of the use of the negative curvature direction produced by the Lanczos method and the effect of the use of the nonmonotone stabilization strategy. To this aim, we have implemented four different algorithms belonging to the class described by the proposed algorithmic model: a monotone algorithm which does not use negative curvature directions, a monotone algorithm which uses negative curvature directions, a nonmonotone algorithm which does not use negative curvature directions and a nonmonotone algorithm which uses negative curvature directions. We have tested and compared them on a large set of standard large scale test problems available from the CUTE collection [2]. The results obtained show that both the use of the negative curvature direction and the nonmonotone stabilization technique play a significant role for improving the performance of the minimization method and indicate clearly that the most efficient algorithm is the one which uses both the negative curvature directions and the nonmonotone stabilization strategy.

Furthermore, in order to have a feel of the effectiveness of our approach, we have compared the results obtained by our best algorithm with those obtained by the default version and the version without preconditioner of the LANCELOT package [3]. The results of these comparisons indicate that the proposed approach is very promising and show that our algorithm is competitive with the LANCELOT algorithms. Even if no final conclusion can be drawn, these numerical results encourage further research

on new large scale minimization algorithms based on the combined use of “sufficiently good” negative curvature directions and nonmonotone stabilization techniques.

2. Globalization algorithm. The aim of this section is to prove that the convergence properties of the nonmonotone stabilization algorithmic model proposed in [8] still hold under weaker assumptions on the negative curvature direction used. We refer to [8, 13, 15, 17] for a detailed discussion of the rationale behind the algorithm. Here we just recall that the approach is based on the following observations:

- The search directions s_k and d_k convey much more information on the minimization problem than single function evaluations, and under standard assumptions, the decrease of their magnitude can be used as an indication that the algorithm is converging.
- The widely used globalization techniques of enforcing the monotone decrease of the objective function values by controlling the magnitude of the step length can deteriorate the behavior of a Newton-type minimization algorithm (see, for example, [13]).

Therefore, the basic idea of the proposed algorithm is to produce the new point by automatically accepting unit step size ($\alpha_k = 1$) in (1.3) without checking the objective function value whenever the magnitude of $\|s_k\| + \|d_k\|$ is decreasing. Otherwise, the algorithm resorts to a nonmonotone linesearch technique to determine the step length.

More in particular, here we describe the steps of the algorithm while we detail the computation of the directions s_k and d_k in the next section.

NONMONOTONE CURVILINEAR STABILIZATION ALGORITHM (NMC).

Data: x_0 , $\Delta_0 > 0$, $\delta \in (0, 1)$, $N \geq 1$, $M \geq 0$, $\sigma \in (0, 1)$ and $\gamma \in (0, \frac{1}{2})$.

Step 1: **Initialization**

Set $k = \ell = j = 0$, $\Delta = \Delta_0$; compute $f(x_0)$ and set $f_0^R = f_0 = f(x_0)$.

Step 2: **Test for convergence**

If $\|g(x_k)\| = 0$ and $H(x_k)$ is positive semidefinite stop.

Step 3: **Tests for automatic step acceptances**

If $k \neq \ell + N$ compute directions s_k and d_k ; then

- if $\|s_k\| + \|d_k\| \leq \Delta$, set $x_{k+1} = x_k + s_k + d_k$, $k = k + 1$, $\Delta = \delta\Delta$ and go to Step 2;
- if $\|s_k\| + \|d_k\| > \Delta$, compute $f(x_k)$ if necessary; then:
 - if $f(x_k) \geq f_j^R$, backtrack to x_ℓ ;
 - otherwise, if $k \neq \ell$ set $\ell = k$, $j = j + 1$, $f_j = f(x_k)$ and update f_j^R according to

$$(2.1) \quad f_j^R = \max_{0 \leq i \leq m(j)} f_{j-i}, \quad \text{where} \quad m(j) = \min[j, M];$$

go to Step 5.

Step 4: **Control after N automatic step acceptances**

If $k = \ell + N$ compute $f(x_k)$ if necessary; then

- if $f(x_k) \geq f_j^R$, backtrack to x_ℓ and go to Step 5;
- if $f(x_k) < f_j^R$, set $\ell = k$, $j = j + 1$, $f_j = f(x_k)$ and update f_j^R according to (2.1); compute directions s_k and d_k ; then:
 - if $\|s_k\| + \|d_k\| \leq \Delta$, set $x_{k+1} = x_k + s_k + d_k$, $k = k + 1$, $\Delta = \delta\Delta$ and go to Step 2;
 - otherwise go to Step 5.

Step 5: Nonmonotone linesearch

Compute $\alpha_k = \sigma^h$ where h is the smallest nonnegative integer such that

$$(2.2) \quad f(x_k + \alpha_k^2 s_k + \alpha_k d_k) \leq f_j^R + \gamma \alpha_k^2 \left[g(x_k)^T s_k + \frac{1}{2} d_k^T H(x_k) d_k \right],$$

set

$$x_{k+1} = x_k + \alpha_k^2 s_k + \alpha_k d_k, \quad k = k + 1, \quad \ell = k, \quad j = j + 1, \quad f_j = f(x_k),$$

update f_j^R according to (2.1) and go to Step 2.

Remark 2.1. We note that, for the sake of simplicity, in the algorithmic scheme we have omitted the explicit indication concerning the necessity to store the point x_ℓ and the directions s_ℓ and d_ℓ whenever we set $\ell = k$.

This algorithmic model derives from the combination of the nonmonotone globalization approach described in [13] and the curvilinear linesearch strategy introduced in [15] and [17]. In particular, if we set $d_k = 0$ for all k in Algorithm NMC, we obtain exactly the algorithmic model of [13], while if we set $\Delta = 0$ and $M = 0$ we obtain the modified Newton methods of [15] and [17].

As we said before, in the algorithm the objective function is not necessarily evaluated at each iteration. We denote by ℓ the index of the last point produced by the algorithm where the objective function has been evaluated; we store the computed objective function values in the sequence $\{f_j\}$ while $\{f_j^R\}$ represent the sequence of *reference values* whose elements are given, at each j , by the maximum over a prefixed number M of previous function values (see (2.1)).

In more detail, in Step 3 we compare the magnitude of $\|s_k\| + \|d_k\|$ with a decreasing sequence of upper bounds in order to validate that the iterates are converging. If this test fails then convergence is enforced by the nonmonotone linesearch procedure. The particular form of this control is motivated by the fact that whenever the sequence $\{x_k\}$ is superlinearly convergent towards a minimizer then $\|s_k\| \rightarrow 0$ and for every $\theta > 0$ and $\delta \in (0, 1)$, the inequality $\|d_k\| \leq \theta \delta^k$ hold for sufficiently large values of k (see [13]). This implies that, eventually, the test at Step 3 is satisfied.

After a predetermined number N of iterations in Step 4, the objective function is computed and compared with the actual reference value in order to evaluate if the new iterate leads to regions where the function is poorly behaved; in this case the algorithm is restarted from the iterate corresponding to the point x_ℓ .

In Step 5 the nonmonotone linesearch procedure is based on the Armijo-type step size rule proposed in [15] with the distinguishing feature that in the right-hand side of (2.2), $f(x_k)$ is replaced by the reference value f_j^R in order to still try to accept the unit step size. For further detail on the nonmonotone stabilization strategy we refer to paper [13].

Now we show that the properties of Algorithm NMC proved in [8] still hold under weaker requirements on the negative curvature direction d_k .

In particular first of all we assume that the directions $\{s_k\}$ and $\{d_k\}$ are bounded. Then we introduce the following conditions:

CONDITION 1. *The directions $\{s_k\}$ are such that $g(x_k)^T s_k \leq 0$ and*

$$g(x_k)^T s_k \rightarrow 0 \quad \text{implies} \quad g(x_k) \rightarrow 0 \quad \text{and} \quad s_k \rightarrow 0.$$

CONDITION 2. *The directions $\{s_k\}$ are such that*

$$s_k \rightarrow 0 \quad \text{implies} \quad g(x_k) \rightarrow 0.$$

CONDITION 3. *The directions $\{d_k\}$ are such that*

$$g(x_k)^T d_k \leq 0, \quad d_k^T H(x_k) d_k \leq 0,$$

$$(2.3) \quad \begin{cases} d_k^T H(x_k) d_k \rightarrow 0, \\ g(x_k) \rightarrow 0 \end{cases} \quad \text{implies} \quad \begin{cases} \min[0, \lambda_{\min}(H(x_k))] \rightarrow 0, \\ d_k \rightarrow 0, \end{cases}$$

where $\lambda_{\min}(H(x_k))$ is the minimum eigenvalue of the Hessian matrix.

Condition 1 is the classical assumption on s_k used in literature to force the global convergence towards a stationary point. In Algorithm NMC (as in the algorithm of [13]) Condition 1 can be replaced with the weaker Condition 2 whenever a new point is accepted without using the linesearch procedure.

Condition 3 concerns the direction d_k and, as Theorem 2.1 will show, is able to ensure the convergence of the algorithm towards stationary points where the Hessian is positive semidefinite. This condition is weaker than those previously used in literature in order to avoid that an algorithm converges toward a point where the Hessian is not positive semidefinite. In fact (see [8, 15, 17, 24]) they require, at least, that (2.3) of Condition 3 be replaced by

$$(2.4) \quad d_k^T H(x_k) d_k \rightarrow 0 \quad \text{implies} \quad \begin{cases} \min[0, \lambda_{\min}(H(x_k))] \rightarrow 0, \\ d_k \rightarrow 0. \end{cases}$$

This difference could appear negligible, but as we said in the introduction, this new condition confirms the fact that the computation of a direction of negative curvature which is a good approximation of an eigenvector corresponding to the most negative eigenvalue is not strongly necessary when we are far from a stationary point.

The following theorem shows that the convergence properties of Algorithm NMC proved in [8] still hold under the weaker conditions stated above. The proof of this theorem is reported in the appendix.

THEOREM 2.1. *Let f be twice continuously differentiable, let x_0 be given, and suppose that the level set $\Omega_0 = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0)\}$ is compact. Assume that the directions s_k and d_k satisfy Conditions 2 and 3 and that, when linesearch is performed, the direction s_k satisfies also Condition 1. Let x_k , $k = 0, 1, \dots$, be the points produced by Algorithm NMC. Then, either the algorithm terminates at some x_p such that $g(x_p) = 0$ and $H(x_p)$ is positive semidefinite or it produces an infinite sequence such that*

(a) *the sequence $\{x_k\}$ remains in a compact set, and every limit point x_* belongs to Ω_0 and satisfies $g(x_*) = 0$. Further, $H(x_*)$ is positive semidefinite and no limit point of $\{x_k\}$ is a local maximum of f ;*

(b) *if the number of stationary points of f in Ω_0 is finite, then the sequence $\{x_k\}$ converges to a stationary point where H is positive semidefinite;*

(c) *if there exists a limit point where H is nonsingular, the sequence $\{x_k\}$ converges to a local minimum point.*

We note that, for simplicity, we stated the preceding theorem under the standard Condition 2; however, the same convergence results can be proved even if we replace Condition 2 with the following weaker condition:

CONDITION 2'. *The directions $\{s_k\}$ and $\{d_k\}$ are such that*

$$\|s_k\| + \|d_k\| \rightarrow 0 \quad \text{implies} \quad g(x_k) \rightarrow 0.$$

Moreover, the proof can be easily modified in a way that the previous convergence results of Theorem 2.1 are ensured under Condition 2 (or Condition 2') and the following slight modifications of Condition 1 and Condition 3.

CONDITION 1'. *The directions $\{s_k\}$ and $\{d_k\}$ are bounded and such that*

$$g(x_k)^T s_k \leq 0, \quad g(x_k)^T d_k \leq 0,$$

$$g(x_k)^T s_k \rightarrow 0 \quad \text{implies} \quad s_k \rightarrow 0,$$

$$g(x_k)^T (s_k + d_k) \rightarrow 0 \quad \text{implies} \quad g(x_k) \rightarrow 0.$$

CONDITION 3'. *The directions $\{d_k\}$ are bounded and such that*

$$g(x_k)^T d_k \leq 0, \quad d_k^T H(x_k) d_k \leq 0,$$

$$\begin{cases} d_k^T H(x_k) d_k \rightarrow 0, \\ g(x_k)^T d_k \rightarrow 0 \end{cases} \quad \text{implies} \quad \begin{cases} \min[0, \lambda_{\min}(H(x_k))] \rightarrow 0, \\ d_k \rightarrow 0. \end{cases}$$

We remark that Condition 1' is weaker than Condition 1, while Condition 3' is stronger than Condition 3.

3. Computation of the search directions. In this section we consider the computation of the search directions s_k and d_k used by Algorithm NMC.

The direction s_k should play a double role; in fact, in addition to enforcing the global convergence of the algorithm, it should also guarantee, under standard assumptions, the superlinear convergence rate. As we are tackling large scale problems, this fact led us to compute the direction s_k by solving approximately the Newton equation

$$(3.1) \quad H(x_k)s = -g(x_k).$$

The most popular approach used to find an approximate solution of (3.1) which satisfies (or can be easily modified to satisfy) Conditions 1 and 2 is the use of the conjugate gradient method (see, for example, [1, 5, 6, 12, 23, 26]).

A related approach to solve approximately (3.1) is the use of the Lanczos algorithm; in fact, in [18, 19], Nash showed that a truncated scheme based on the Lanczos algorithm allows us to obtain an effective Newton-type direction.

As regards the direction d_k , it should help the algorithm to escape from the region where the objective function is nonconvex. Furthermore, Condition 3 indicates that its contribution is more necessary as the gradient of the objective function is smaller in this region. Therefore, the direction d_k should be a nonascent direction of negative curvature such that it has a resemblance with an eigenvector of the Hessian matrix corresponding to the most negative eigenvalue, and this resemblance should be stronger as the gradient is smaller.

As indicated in [24] (see also [22]), a “good” approximation of the eigenvector corresponding to the smallest eigenvalue of the Hessian matrix can be efficiently obtained by using the Lanczos algorithm.

Therefore, on the basis of these considerations, a natural choice for us is that of using a truncated scheme based on the Lanczos algorithm for computing both search directions used in our algorithm.

For the sake of completeness, now we describe the general scheme of the Lanczos algorithm with starting vector $q_1 = -g(x_k)$ (see, for example, [4, 10, 22]).

LANCZOS ALGORITHM.

Data: $q_1 = -g(x_k)$

Step 1: $i = 1$, $v_0 = 0$, $\beta_i = \|q_i\|$

Step 2: $v_i = \frac{1}{\beta_i} q_i$, $\alpha_i = v_i^T H(x_k) v_i$, $q_{i+1} = H(x_k) v_i - \alpha_i v_i - \beta_i v_{i-1}$, $\beta_{i+1} = \|q_{i+1}\|$

Step 3: If $\|\beta_{i+1}\| = 0$ stop; otherwise $i = i + 1$ and go to Step 2.

At the i th iteration the following matrices can be defined:

$$V_i = \begin{pmatrix} v_1 & v_2 & \vdots & \vdots & v_i \end{pmatrix}, \quad T_i = \begin{pmatrix} \alpha_1 & \beta_2 & 0 & \dots & 0 \\ \beta_2 & \alpha_2 & \ddots & 0 & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & 0 & \ddots & \alpha_{i-1} & \beta_i \\ 0 & \dots & 0 & \beta_i & \alpha_i \end{pmatrix}.$$

Now we recall some important results concerning the Lanczos algorithm which are summarized in the following theorem (see [4, 10, 22]).

THEOREM 3.1.

(i) If $g(x_k)$ has nonzero projections only on p eigenvectors of $H(x_k)$ and if the distinct eigenvalues corresponding to these p eigenvectors are $m \leq p$, then the Lanczos algorithm terminates after m steps with $\beta_{m+1} = 0$ and $\beta_i \neq 0$, $i = 1, \dots, m$.

(ii) Suppose that $\tilde{y}_i$ is the solution of the system $T_i y = -V_i^T g(x_k)$, and set

$$(3.2) \quad \tilde{s}_i = V_i \tilde{y}_i.$$

Then, if $\beta_{i+1} = 0$, the vector $\tilde{s}_i$ is a solution of the system $H(x_k)s = -g(x_k)$; otherwise, if $\beta_{i+1} \neq 0$, the vector $\tilde{s}_i$ is the solution of the system $V_i^T (H(x_k)s + g(x_k)) = 0$.

(iii) Assume that $H(x_k)$ has m distinct eigenvalues and that $g(x_k)$ has nonzero projections on m eigenvectors of $H(x_k)$ which correspond to distinct eigenvalues. Let μ_m be the minimum eigenvalue of the tridiagonal matrix T_m produced by the Lanczos algorithm, and let w_m be the corresponding eigenvector. Then μ_m is also the minimum eigenvalue of $H(x_k)$, and its corresponding eigenvector is given by

$$(3.3) \quad \tilde{d}_m = V_m w_m.$$

(iv) If μ_i is the minimum eigenvalue of the tridiagonal matrix T_i , then there exists an eigenvalue λ of $H(x_k)$ such that $\|\lambda - \mu_i\| \leq \beta_{i+1}$.

(v) Assume that λ is the smallest eigenvalue of $H(x_k)$ and μ_i is the smallest eigenvalue of the tridiagonal matrix T_i . Then

$$\lambda \leq \mu_i \leq \lambda + \frac{C}{\phi_{i-1}^2},$$

where C is a positive scalar and ϕ_{i-1} is the $i-1$ degree Chebyshev polynomial.

Property (i) of Theorem 3.1 ensures that the algorithm terminates in a finite number of steps with $\beta_{m+1} = 0$. Property (ii) shows that the Lanczos algorithm can be successfully used for approximately solving the Newton equation (3.1).

Properties (iii), (iv), and (v) of Theorem 3.1 show that the Lanczos algorithm can be a useful tool for efficiently computing a good approximation of the most negative eigenvalue and the corresponding eigenvectors of the Hessian matrix $H(x_k)$. In particular, property (iii) shows that in the case of termination with $\beta_{m+1} = 0$ the minimum eigenvalue and the corresponding eigenvector of the Hessian matrix $H(x_k)$ can be obtained by computing the minimum eigenvalue and the corresponding eigenvector of the tridiagonal matrix T_m and, as is well known, this last computation can be done in a very efficient way by taking the particular structure of the matrix T_m into account. Property (iv) provides a simple bound on the accuracy of the estimate of the smallest eigenvalue of the Hessian matrix obtained at the i th iteration of Lanczos procedure. Finally, property (v) shows that the smallest eigenvalue of the tridiagonal matrix T_i approaches the smallest eigenvalue of the Hessian matrix very quickly.

These considerations show that, at each iteration, the Lanczos algorithm produces two vectors $\tilde{s}_i$ and $\tilde{d}_i = V_i w_i$ which are approximations of a Newton-type direction and of an eigenvector corresponding to the smallest eigenvalue of the Hessian matrix $H(x_k)$, respectively. Furthermore, we can easily check the accuracy of these approximations. However, the use of (3.2) and (3.3) for the computation of $\tilde{s}_i$ and $\tilde{d}_i$ requires the storage of the matrix V_i , which limits their applications when i is large. Fortunately, this drawback can be efficiently overcome in the computation of the vector $\tilde{s}_i$. In fact, in [21] it is shown that the properties of the Lanczos algorithm, the particular choice of the starting vector $q_1 = -g(x_k)$, and the particular structure of the tridiagonal matrix T_i allow us to compute $\tilde{s}_i$ from the estimate $\tilde{s}_{i-1}$ obtained at the previous iteration by using update formulae which do not require any storage and matrix computations. In particular in our algorithm we use the SYMMLQ method proposed in [21], which is considered the most theoretically and numerically satisfactory method for solving linear systems involving indefinite symmetric matrices. Another feature of this solver of linear systems is that in the SYMMLQ algorithm, the Lanczos iterates are not interrupted even if the system is not consistent. This is particularly appealing in our approach since, also in this case, we can obtain useful information for computing a “good” direction of negative curvature d_k . In fact, in particular, we could compute the vector $\tilde{d}_i$ which is closely related to the eigenvector corresponding to the minimum eigenvalue of the Hessian matrix and use it as direction d_k . Unfortunately, the use of the matrix V_i whose columns are the Lanczos vectors $v_1, \dots, v_i$ cannot be avoided in the computation of $\tilde{d}_i$ given by (3.3). Therefore, due to the requirement of limited storage room, we store only a fixed number L of Lanczos vectors in order to use a matrix V_L of reasonable dimensions. Property (v) of Theorem 3.1 seems to indicate that few Lanczos vectors are needed to ensure that the direction $\tilde{d}_L$ has enough resemblance to the eigenvector corresponding to the minimum eigenvalue of the Hessian matrix.

Taking the previous considerations into account, we have based the computation of the directions d_k and s_k used in our algorithm NMC on the SYMMLQ routine by Paige and Saunders [21]. In particular we can roughly summarize the scheme that we have used for computing both search directions in the following steps:

- *Lanczos iterations.*

The SYMMLQ routine has been slightly modified in a way that it continues to perform Lanczos iterations until the i th iteration, where *both*

- one of the stopping criteria of the original version of SYMMLQ is satisfied,
- and $|\beta_{i+1}| < \epsilon_k$ or $i > L$.

- *Computation of direction s_k .*

- If the system (3.1) is solvable and the vector $\tilde{s}_i$ produced by SYMMLQ routine satisfies Conditions 1 and 2, then set $s_k = \tilde{s}_i$.
- Otherwise, set $s_k = -g(x_k)$.
- *Computation of direction d_k .*
 Compute the most negative eigenvalue μ_i of the tridiagonal matrix T_i ;
 - If $\mu_i < 0$, then compute the corresponding eigenvector w_i , and set

$$d_k = -\rho_k \operatorname{sgn} \left[g(x_k)^T \tilde{d}_i \right] \tilde{d}_i,$$

where $\tilde{d}_i = V_i w_i$ and ρ_k is a suitable scaling factor.

- Otherwise, set $d_k = 0$.

Remark 3.1. We note that if enough storage room was available, we could set $L = n$, and hence, the previous scheme could compute a direction of negative curvature d_k that ensures the convergence of the algorithm NMC toward stationary points where the Hessian matrix is positive semidefinite. More precisely, when enough storage is available and the assumption required at part (iii) of Theorem 3.1 is fulfilled, then Condition 3 is satisfied and Theorem 2.1 applies; otherwise, it is not ensured.

Remark 3.2. We point out that some of the choices concerning the search directions are dictated by the need of simplifying the analysis and to concentrate on the issues related to the main topic of the paper, i.e., the study of the possibility as well as the importance of exploiting the information on the local nonconvexity of the objective function contained in the second order derivatives. In particular, the use of the steepest descent direction as “alternative” choice of the direction s_k is the simplest one. More efficient choices are certainly possible; for example, the global convergence can be ensured also by using more sophisticated directions obtained by minimizing the quadratic model of the objective function over the subspace spanned by the positive curvature directions produced by the Lanczos algorithm. As regards the negative curvature direction used in the algorithm, as already pointed out in the introduction, our choice has been motivated by its theoretical properties which guarantee a good resemblance to the eigenvector corresponding to the smallest eigenvalue of the Hessian matrix. Actually, other negative curvature directions can be obtained as a simple by-product of the Lanczos algorithm; these directions, even if very simple to compute, need further investigation to understand if they act sufficiently like the eigenvector corresponding to the smallest eigenvalue of the Hessian matrix. All these possibilities must be investigated and exploited in order to design an efficient code. This is beyond the aim of this paper and will be the subject of future work.

4. Numerical experiences. In this section we report the results of extensive numerical testing performed on all the unconstrained large scale problems available from the CUTE collection [2]. This test set consists of 98 functions covering all the classical test problems along with a large number of nonlinear optimization problems of various difficulty. The problems we have solved range from problems with 1000 variables to problems with 10,000 variables.

We made all the runs on an IBM RISC System/6000 375 using Fortran in double precision with the default optimization compiling option. For all the algorithms considered, the termination criterion is $\|g\| \leq 10^{-5}$. In the sequel, when we compare the results of different algorithms we consider all the test problems where the algorithms converge to the same point; moreover, in these comparisons the results of two runs are reputed equal if they differ by at most 5%.

The first purpose of these numerical experiences is to investigate the influence of

the presence of the negative curvature directions and the influence of the nonmonotone stabilization technique. To this aim, we have implemented four different algorithms belonging to the proposed class:

1. a monotone algorithm which does not use negative curvature directions (**Mon**);
2. a monotone algorithm which uses negative curvature directions (**MonNC**);
3. a nonmonotone algorithm which does not use negative curvature directions (**NMon**);
4. a nonmonotone algorithm which uses negative curvature directions (**NMonNC**).

The second purpose is to have an idea of the potentiality of the approach described in this work in terms of efficiency and robustness. In particular, we have compared the results of our best algorithm with those obtained by the default band preconditioner version and the version without preconditioner of the LANCELOT package [3].

Of course, an efficient implementation of an algorithm belonging to the proposed class should be based on extensive empirical tuning of all the parameters which appear in the algorithm. Since the definition of an efficient code is not a goal of this paper, we have not adjusted all the parameters very accurately. In most of the cases, we have adopted (or slightly modified) choices already proposed in literature. The accurate analysis of the sensitivity of the algorithm as the parameters vary will be the subject of future work.

In more detail, as regards the stabilization scheme of the two nonmonotone algorithms, we adopt essentially the same parameters as in [13] with some modifications in order to take the fact that we are considering large scale problems into account; in particular, unlike [13], we use the following values: $M = 20$, $N = 20$, and $\Delta_0 = 10^3$. For the two monotone algorithms we set $M = 0$ and $\Delta_0 = 0$. The only difference with respect to the implementation described in [13] is that, following [28], we choose to reduce the freedom of the algorithm whenever a backtracking is performed; in fact, we set $\Delta = 10^{-1}\Delta$ and $M = M/5 + 1$ whenever the algorithm backtracks to the last accepted point where the function has been evaluated.

In the scheme of the computation of the search directions we have set the user-specified tolerance required by the SYMMLQ routine to the value

$$rtol = \tau \min \left\{ 1, \|g(x_k)\| \max \left(\frac{1}{k+1}, \frac{1}{\exp\left(\frac{k}{\tau n}\right)} \right) \right\},$$

where $\tau = 10^{-1}$. The presence of the term $1/\exp\left(\frac{k}{\tau n}\right)$ reflects the fact that we are tackling large scale problems.

In the two algorithms which use negative curvature directions, we have set

$$(4.1) \quad \epsilon_k = 10^3 \max \left\{ \|g(x_k)\|, \frac{1}{k+1} \right\}$$

and

$$(4.2) \quad \rho_k = \sqrt{-\mu_i} \min \left\{ 1, \frac{1}{\|g_k\|} \right\},$$

where μ_i is the smallest eigenvalue of the matrix T_i produced by the Lanczos algorithm. The rationale behind the choice (4.1) is to force the parameter ϵ_k to follow the behavior of the magnitude of the gradient only when k is large. The particular choice

TABLE 4.1
Cumulative results for **NMonNC** with different values of L .

	$L = 10$	$L = 50$	$L = 100$	$L = 200$	$L = 500$
Iterat.	6206	5054	4884	4897	4786
Funct.	6378	5369	5243	5488	5303
Grad.	6304	5152	4982	4995	4884
Time	4934.05	3846.99	4067.2	6131.46	8319.56

(4.2) of the scaling factor ρ_k derives from the one proposed by Moré and Sorensen in [17], where we added the term $\min\{1, 1/\|g_k\|\}$ since numerical experiences have indicated that it is better not to perturb the Newton-type direction too much when the iterates are far from a stationary point.

The complete results of our numerical experiences are reported in [14], and here, for sake of brevity, we report some summaries of this extensive numerical testing.

First we focused our attention more on the parameter L , which plays a crucial role for the applicability of our approach. We recall that this parameter represents the total number of Lanczos basis vectors stored; this number must be fixed to a small prescribed value in order to limit the storage requirements and, on the other hand, it influences the “goodness” of the negative curvature direction. Of course, this parameter appears only in the algorithms **MonNC** and **NMonNC**, which use negative curvature directions.

Since the effect in these two algorithms is quite similar, for the sake of simplicity, in Table 4.1 we report the cumulative results (that is the total number of iterations, function and gradient evaluations, and total CPU time, in seconds) obtained by algorithm **NMonNC** with different values L on the whole test set of 98 test problems.

On the basis of these results, both the values $L = 50$ and $L = 100$ seem to be acceptable; in fact, for values $L \geq 50$ the influence of the negative curvature directions can be observed, but after this value an increase of CPU time is clearly shown without any substantial improvement of the efficiency of the algorithm. Hence, in what follows we have chosen as default value $L = 50$.

Another significant investigation concerns how expensive the computations of the most negative eigenvalue μ_i of the tridiagonal matrix T_i and of its corresponding eigenvector w_i are. In the algorithms **NMonNC** and **MonNC**, which require the computation of μ_i and w_i in order to determine the directions of negative curvature d_k , we have computed “exactly” μ_i and w_i by using a QR factorization of the tridiagonal matrix T_i which exploits the particular structure of this matrix. The numerical results show that the computational burden for the algorithm **NMonNC** (similar situation is for **MonNC**) for determining the eigenvalue μ_i and the corresponding eigenvector w_i is negligible with respect to the time needed to approximately solve the Newton equation. In fact, the cumulative CPU time needed for solving the Newton equation is 3649.19 seconds, while for computing μ_i and w_i it is 17.13 seconds. We remark that in our framework the computation of μ_i and w_i could be performed by some truncated scheme; however, on the basis of these results, this less expensive truncated computation seems not to be necessary at least for the problems considered.

Now, we report some summaries of the results obtained by the four algorithms **Mon**, **MonNC**, **NMon**, and **NMonNC**. In the comparisons of the performances of these algorithms reported in Tables 4.2 and 4.3 we have not considered the test problems BROYDEN7D and EIGENALS since the four algorithms converge to different points. In particular, in Table 4.2 we report how many times each of the four algorithms is the best, second, third, and worst, and in Table 4.3 we give the cumulative

TABLE 4.2

Comparative ranking for the algorithms Mon, MonNC, NMon, and NMonNC.

	Algorithm	1st	2nd	3rd	4th
Iterations	Mon	78	2	14	2
	MonNC	80	0	16	0
	NMon	77	2	10	7
	NMonNC	79	4	13	0
Functions Evaluations	Mon	5	6	78	7
	MonNC	5	3	79	9
	NMon	87	8	1	0
	NMonNC	80	15	1	0
Gradients Evaluations	Mon	78	2	15	1
	MonNC	80	0	15	1
	NMon	77	2	10	7
	NMonNC	79	4	13	0
CPU time	Mon	50	13	32	1
	MonNC	35	13	40	8
	NMon	76	10	5	5
	NMonNC	77	9	10	0

TABLE 4.3

Cumulative results for the algorithms Mon, MonNC, NMon, and NMonNC.

	Mon Monotone without neg. curv.	MonNC Monotone with neg. curv.	NMon Nonmonotone without neg. curv.	NMonNC Nonmonotone with neg. curv.
Iterations	5693	5077	5466	4793
Funct. eval.	8690	8163	5324	5091
Grad. eval.	5789	5173	5562	4889
CPU time	6444.09	6376.21	4540.92	3674.77

TABLE 4.4

The obtained objective function values of test problems BROYDEN7D and EIGENALS.

	Mon	MonNC	NMon	NMonNC
BROYDN7D	369.225	368.240	378.249	368.223
EIGENALS	485.00	.608738D-9	1327.00	.162078D-9

results.

As can be easily seen from these tables, algorithms **MonNC** and **NMonNC** are the best in terms of number of iterations and gradient evaluations, while **NMon** and **NMonNC** are very clearly the best in terms of number of function evaluations and CPU time. This indicates that the use of negative curvature directions has a beneficial effect in terms of number of iterations and gradient evaluations. On the other hand, the effect of the nonmonotone strategy is especially evidenced as regards the number of function evaluations and CPU time.

Another indication of the importance of the negative curvature directions is suggested by the results obtained on the test problems EIGENALS and BROYDN7D, which we did not consider in the previous comparison. In fact, Table 4.4, where we report the best objective function value obtained by the four algorithms, shows that on these problems the use of negative curvature directions enables both algorithms **MonNC** and **NMonNC** to converge toward “better points,” i.e., points where the objective function value is lower.

In conclusion, the results of Tables 4.2, 4.3, and 4.4 indicate that the best algorithm is **NMonNC** since the simultaneous use of both the negative curvature

TABLE 4.5

Number of times each method performs the best on 97 test problems.

	NMonNC	LANCELOT default	tie
Iterations	50	41	6
Function evaluations	72	19	6
Gradient evaluations	44	41	12
CPU time	65	29	3

TABLE 4.6

*Statistic of rankings of the wins for **NMonNC** and LANCELOT default.*

	Iterations			
	5%–25%	25%–50%	50%–75%	75%–100%
NMonNC	15	17	8	10
LANCELOT default	13	9	13	6

	Function Evaluations			
	5%–25%	25%–50%	50%–75%	75%–100%
NMonNC	2	24	22	24
LANCELOT default	2	9	2	6

	Gradient Evaluations			
	5%–25%	25%–50%	50%–75%	75%–100%
NMonNC	15	16	6	7
LANCELOT default	10	19	6	6

	CPU Time			
	5%–25%	25%–50%	50%–75%	75%–100%
NMonNC	9	13	20	23
LANCELOT default	6	5	3	15

TABLE 4.7

*Cumulative results for **NMonNC** and LANCELOT default.*

	NMonNC	LANCELOT default
Iterations	4964	6718
Function evaluations	5209	6815
Gradient evaluations	5061	5831
CPU time	3840.17	9471.09

directions and the nonmonotone strategy allows a substantial improvement in terms of number of iterations, function and gradient evaluations, CPU time, and a tendency to converge towards “better” stationary points.

In the sequel we report the comparisons between the results of our best algorithm **NMonNC** and those obtained by the LANCELOT software package [3]; in particular, we have considered the default LANCELOT band preconditioner and the version without preconditioner. We excluded from these comparisons the test problem BROYDN7D, where the two LANCELOT algorithms converge to different points with respect to **NMonNC** algorithm.

In Table 4.5 we report the number of wins in terms of number of iterations, function and gradient evaluations, and CPU time of the new algorithm **NMonNC** and the default LANCELOT band preconditioner on the test set, while in Table 4.6 we show the distribution of these wins (namely, how many times the percent gain of each algorithm is between 5% and 25%, 25% and 50%, 50% and 75%, 75% and 100%). Finally, in Table 4.7 we summarize the cumulative results obtained by these

TABLE 4.8

CPU time for solving the test problems FMINSURF 5625 and EIGENBLS.

	NMonNC	LANCELOT default
FMINSURF 5625	81.01	2503.00
EIGENBLS 930	1465.42	3338.57
Total	1546.43	5841.57

TABLE 4.9

Cumulative results without problem FMINSURF 5625 and EIGENBLS.

	NMonNC	LANCELOT default
Iterations	4628	5891
Function evaluations	4551	5986
Gradient evaluations	4723	5155
CPU time	2293.74	3629.52

TABLE 4.10

Number of times each method performs the best on 97 test problems.

	NMonNC	LANCELOT without precondition.	tie
Iterations	47	42	8
Function evaluations	76	14	7
Gradient evaluations	38	47	12
CPU time	76	19	2

algorithms. The reported results clearly show the good behavior of the new algorithm **NMonNC** and the relevant computational savings. It is important to notice, as the total CPU time required by our new algorithm **NMonNC** to solve all the problems is reduced more than 50%.

However, it can be pointed out that the great difference between the cumulative time required by the two algorithms is mainly due to the particular efficiency of our method on the two test problems FMINSURF 5625 and EIGENBLS, as is shown in Table 4.8. However, Algorithm **NMonNC** still outperforms LANCELOT even if we do not consider these problems in the computation of the cumulative results (see Table 4.9).

Now, in the tables that will follow, we report the results of the comparison between our algorithm **NMonNC** and LANCELOT without preconditioner. In particular, Table 4.10 summarizes the number of wins of these two algorithms on the test set, and Table 4.11 reports the distribution of their wins.

From Table 4.10 we can note that the version of LANCELOT without preconditioner appears efficient with regard to the number of gradient evaluations, but it is greatly beaten with regard to the wins in terms of CPU times and number of function evaluations. Table 4.11, besides confirming the superiority of **NMonNC** with regard to wins in terms of number of iterations, number of function evaluations, and CPU times, shows that most of the wins of LANCELOT without preconditioner in terms of number of gradient evaluations produce moderate gains. The cumulative results of our **NMonNC** and LANCELOT without preconditioner (see Table 4.12) show that **NMonNC** outperforms LANCELOT without preconditioner also in terms of gradient evaluation.

However, by observing the detailed results, it can be seen that the large number of gradient evaluations required by the version without preconditioner of LANCELOT is essentially due to the excessive number of gradient evaluations (1458 for **NMonNC**

TABLE 4.11

Statistic of rankings of the wins for NMonNC and LANCELOT without preconditioner.

	Iterations			
	5%–25%	25%–50%	50%–75%	75%–100%
NMonNC	20	13	6	8
LANCELOT without prec.	15	18	6	3

	Function Evaluations			
	5%–25%	25%–50%	50%–75%	75%–100%
NMonNC	7	26	22	21
LANCELOT without prec.	6	3	3	2

	Gradient Evaluations			
	5%–25%	25%–50%	50%–75%	75%–100%
NMonNC	17	10	4	7
LANCELOT without prec.	18	20	6	3

	CPU Time			
	5%–25%	25%–50%	50%–75%	75%–100%
NMonNC	16	30	12	18
LANCELOT without prec.	8	3	5	3

TABLE 4.12

Cumulative results for NMonNC and LANCELOT without preconditioner.

	NMonNC	LANCELOT without prec.
Iterations	4964	8026
Function evaluations	5209	8123
Gradient evaluations	5061	31663
CPU time	3840.17	4692.86

and 27791 for LANCELOT) needed for solving the test problem FLETCHCR 1000.

However, even if we do not consider this function, a significant computational savings is still noticeable (see Table 4.13).

On the basis of these results, it can be easily argued that our method outperforms the behavior of both the default band preconditioner algorithm and the version without preconditioner implemented in LANCELOT since a considerable reduction of the computational burden and an additional robustness are shown in both cases. This computational saving is more evident in terms of iterations, number of function evaluations, and CPU time.

5. Conclusions. In this paper we have considered a nonmonotone stabilization algorithmic model based on a curvilinear linesearch, which uses a combination of a Newton-type direction and a negative curvature direction. In particular we have shown that, under weak assumptions on the negative curvature direction, this algorithmic model produces a sequence of points converging toward stationary points which satisfy the second order necessary optimality condition. This new theoretical result gives useful hints on how to define new efficient large scale algorithms. In particular, we propose a class of algorithms which use the Lanczos method for computing, at each iteration, both a truncated Newton direction and a negative curvature direction.

The results obtained by a preliminary numerical experimentation indicate that the use of the nonmonotone strategy is always very effective both if it is used in an algorithm which does not use negative curvature directions and if it is used in an algorithm which uses negative curvature directions. With regard to the use of

TABLE 4.13
Cumulative results without FLETCHCR 1000 test problem.

	NMonNC	LANCELOT
Iterations	3507	4507
Function evaluations	3447	4603
Gradient evaluations	3603	3872
CPU time	3751.23	4510.78

negative curvature directions, our numerical experiences show that

- the use of negative curvature directions in a monotone algorithm leads to an improvement in terms of number of iterations and function and gradient evaluations but no significant difference in terms of CPU time;
- the benefits of the use of negative curvature directions are very evident in a nonmonotone context—in fact, in this case, good improvements in terms of CPU time can also be obtained.

The conclusion could be that the potentiality of negative curvature directions in a linesearch approach can be fully exploited by using nonmonotone strategies.

These computational results show also that our method outperforms the default band preconditioner version and the version without preconditioner of the LANCELOT package both in terms of computational burden and in terms of robustness. Even if no final conclusion can be drawn, this seems to indicate that the proposed approach is reliable and promising for defining new efficient algorithms for large scale minimization problems.

Another point which must be taken into account in comparing our approach with the one of the LANCELOT algorithms is the price that our algorithm must pay for using more sophisticated negative curvature directions. In fact, the computation of this direction requires the storage of an $n \times L$ matrix whose columns are the first L Lanczos basis vectors produced by SYMMLQ routine. Even if relatively small values of the parameter L are sufficient to ensure good behavior of our algorithm, the fact of requiring no matrix storage is an unquestionable advantage of the LANCELOT algorithms. Therefore, in conclusion, we think that new very efficient algorithms for large scale minimization problems could be defined by trying to join the advantages of our approach with those of the LANCELOT approach. In particular, a challenging research argument is the computation of good negative curvature directions which include enough second order information contained in the Hessian matrix without using matrix storage. As future work, we believe that promising approaches could be a smarter utilization of the information obtained during the conjugate gradient iterates and the use of a new efficient version of the Lanczos algorithm (see, for example, [25]).

Appendix. In this appendix we report some technical lemmas and the proof of Theorem 2.1. First of all we introduce some notations used in the following: we denote by $\{x_{\ell(j)}\}$ the sequence of points produced by Algorithm NMC where the objective function is evaluated, and we indicate with $\hat{f}_k^R$ the value of the reference value f_j^R at the k th iteration of the algorithm.

LEMMA A.1 (see [8]). *Assume that Algorithm NMC produces an infinite sequence $\{x_k\}$; then*

- (a) *the sequence $\{f_j^R\}$ is nonincreasing and has a limit f_*^R ;*

(b) for any index j we have

$$f_i^R < f_j^R \quad \text{for all } i > j + M;$$

that is, the reference value f_j^R must decrease after at most $M+1$ function evaluations;

(c) the sequence $\{x_k\}$ remains in a compact set.

LEMMA A.2 (see [8]). Assume that Algorithm NMC produces an infinite sequence $\{x_k\}$. Then we can thin the sequence $\{x_{\ell(j)}\}$ so that it satisfies the following conditions:

- (a) $f_j^R = f(x_{\ell(j)})$ for $j = 0, 1, \dots$;
- (b) for any integer k , there exists an index j_k such that

$$0 < \ell(j_k) - k \leq N(M+1), \quad f_{j_k}^R = f(x_{\ell(j_k)}) < \hat{f}_k^R.$$

LEMMA A.3. Assume that Algorithm NMC produces an infinite sequence $\{x_k\}$ and that the assumptions of Theorem 2.1 hold. Then we have

- (a) $\lim_{k \rightarrow \infty} f(x_k) = \lim_{k \rightarrow \infty} \hat{f}_k^R = \lim_{j \rightarrow \infty} f_j^R = f_*^R$;
- (b) $\lim_{k \rightarrow \infty} \alpha_k^2 \|s_k\| = 0$, $\lim_{k \rightarrow \infty} \alpha_k \|d_k\| = 0$, implying that

$$\lim_{k \rightarrow \infty} \|x_{k+1} - x_k\| = 0.$$

Proof. Note that in Step 3 (a) and Step 4 (b) of Algorithm NMC we may accept a step without performing a linesearch. Let $\{x_k\}_{\mathcal{L}}$ denote the set of points where a linesearch is performed. Then

$$(A.1) \quad \|s_k\| + \|d_k\| \leq \Delta_0 \delta^t \quad \text{for } k \notin \mathcal{L},$$

where the integer t increases with $k \notin \mathcal{L}$; when $k \notin \mathcal{L}$, we set $\alpha_k = 1$ for convenience. It follows from (A.1) that if we do not perform a linesearch an infinite number of times, then

$$(A.2) \quad \lim_{\substack{k \rightarrow \infty \\ k \notin \mathcal{L}}} \alpha_k^2 \|s_k\| = 0 \quad \text{and} \quad \lim_{\substack{k \rightarrow \infty \\ k \notin \mathcal{L}}} \alpha_k \|d_k\| = 0.$$

We show by induction that, for any fixed integer $i \geq 1$, we have

$$(A.3) \quad \lim_{j \rightarrow \infty} \alpha_{\ell(j)-i}^2 \|s_{\ell(j)-i}\| = 0 \quad \lim_{j \rightarrow \infty} \alpha_{\ell(j)-i} \|d_{\ell(j)-i}\| = 0,$$

and

$$(A.4) \quad \lim_{j \rightarrow \infty} f(x_{\ell(j)-i}) = \lim_{j \rightarrow \infty} f(x_{\ell(j)}) = \lim_{j \rightarrow \infty} f_j^R = f_*^R.$$

(Here and in what follows we assume that the index j is large enough to avoid the occurrence of negative subscripts.) Assume first that $i = 1$, and consider two subsequences of $\{\ell(j)-1\}$, corresponding to whether $\ell(j)-1$ is in $\mathcal{L}$ or not. If either of these subsequences is finite, then we can discard the corresponding elements. Otherwise, for $\ell(j)-1 \notin \mathcal{L}$, then (A.3) holds with $i = 1$. Now consider the other subsequence, where $\ell(j)-1 \in \mathcal{L}$. Recalling the acceptability criterion of the nonmonotone linesearch, and by Lemma A.2 (a) we have

$$\begin{aligned} f_j^R &= f(x_{\ell(j)}) = f\left(x_{\ell(j)-1} + \alpha_{\ell(j)-1}^2 s_{\ell(j)-1} + \alpha_{\ell(j)-1} d_{\ell(j)-1}\right) \\ &\leq \hat{f}_{\ell(j)-1}^R + \gamma \alpha_{\ell(j)-1}^2 \left[g(x_{\ell(j)-1})^T s_{\ell(j)-1} + \frac{1}{2} d_{\ell(j)-1}^T H(x_{\ell(j)-1}) d_{\ell(j)-1} \right]. \end{aligned}$$

It follows that

$$(A.5) \quad \hat{f}_{\ell(j)-1}^R - f_j^R \geq \gamma \alpha_{\ell(j)-1}^2 \left| g(x_{\ell(j)-1})^T s_{\ell(j)-1} + \frac{1}{2} d_{\ell(j)-1}^T H(x_{\ell(j)-1}) d_{\ell(j)-1} \right|.$$

Therefore, if $\ell(j) - 1 \in \mathcal{L}$ for an infinite subsequence, from Lemma A.1(a) and (A.5), and the observation that $g(x_{\ell(j)-1})^T s_{\ell(j)-1} \leq 0$ and $d_{\ell(j)-1}^T H(x_{\ell(j)-1}) d_{\ell(j)-1} \leq 0$, we get

$$(A.6) \quad \alpha_{\ell(j)-1}^2 g(x_{\ell(j)-1})^T s_{\ell(j)-1} \rightarrow 0 \quad \text{and} \quad \alpha_{\ell(j)-1}^2 d_{\ell(j)-1}^T H(x_{\ell(j)-1}) d_{\ell(j)-1} \rightarrow 0.$$

Now, (A.6) implies that either $\alpha_{\ell(j)-1} \rightarrow 0$ or

$$\begin{cases} g(x_{\ell(j)-1})^T s_{\ell(j)-1} \rightarrow 0, \\ d_{\ell(j)-1}^T H(x_{\ell(j)-1}) d_{\ell(j)-1} \rightarrow 0. \end{cases}$$

In the first case we have $\alpha_{\ell(j)-1}^2 \|s_{\ell(j)-1}\| \rightarrow 0$ and $\alpha_{\ell(j)-1} \|d_{\ell(j)-1}\| \rightarrow 0$. In the second case, by Condition 1 we have $\alpha_{\ell(j)-1}^2 \|s_{\ell(j)-1}\| \rightarrow 0$ (taking $\alpha_k \leq 1$). Condition 1 also implies $g(x_{\ell(j)-1}) \rightarrow 0$, and hence, by Condition 3, we obtain $\alpha_{\ell(j)-1} \|d_{\ell(j)-1}\| \rightarrow 0$.

It can be concluded that (A.3) holds for $i = 1$. Moreover, since

$$f(x_{\ell(j)}) = f(x_{\ell(j)-1} + \alpha_{\ell(j)-1}^2 s_{\ell(j)-1} + \alpha_{\ell(j)-1} d_{\ell(j)-1}),$$

(A.3) and the uniform continuity of f on the compact set containing $\{x_k\}$ imply that (A.4) also holds for $i = 1$.

Assume now that (A.3) and (A.4) hold for a given $i \geq 1$, and consider the point $x_{\ell(j)-i-1}$. Reasoning as before, we can again distinguish between the case $\ell(j) - i - 1 \notin \mathcal{L}$, when (A.1) holds with $k = \ell(j) - i - 1$, and the case $\ell(j) - i - 1 \in \mathcal{L}$, in which we have

$$(A.7) \quad \begin{aligned} & \hat{f}_{\ell(j)-i-1}^R - f(x_{\ell(j)-i}) \\ & \geq \gamma \alpha_{\ell(j)-i-1}^2 \left| g(x_{\ell(j)-i-1})^T s_{\ell(j)-i-1} + \frac{1}{2} d_{\ell(j)-i-1}^T H(x_{\ell(j)-i-1}) d_{\ell(j)-i-1} \right|. \end{aligned}$$

Using (A.2), (A.4), and (A.7) and recalling Conditions 1 and 3 we can assert that equations (A.3) hold with i replaced by $i + 1$. By (A.3) and the uniform continuity of f , it follows that (A.4) is also satisfied with i replaced by $i + 1$, which completes the induction.

Now, let x_k be any given point produced by the algorithm. Then by Lemma A.2 there is an index j_k such that

$$(A.8) \quad 0 < \ell(j_k) - k \leq N(M + 1).$$

Then we can write

$$x_k = x_{\ell(j_k)} - \sum_{i=1}^{\ell(j_k)-k} \left[\alpha_{\ell(j_k)-i}^2 s_{\ell(j_k)-i} + \alpha_{\ell(j_k)-i} d_{\ell(j_k)-i} \right],$$

and this implies, by (A.3) and (A.8), that

$$\lim_{k \rightarrow \infty} \|x_k - x_{\ell(j_k)}\| = 0.$$

It follows from the uniform continuity of f that

$$(A.9) \quad \lim_{k \rightarrow \infty} f(x_k) = \lim_{k \rightarrow \infty} f(x_{\ell(j_k)}) = \lim_{j \rightarrow \infty} f_j^R = \lim_{k \rightarrow \infty} \hat{f}_k^R,$$

and (a) is proved.

If $k \in \mathcal{L}$, we obtain $f(x_{k+1}) \leq \hat{f}_k^R + \gamma \alpha_k^2 g(x_k)^T s_k + \frac{\gamma}{2} \alpha_k^2 d_k^T H(x_k) d_k$, and hence we have that

$$(A.10) \quad \hat{f}_k^R - f(x_{k+1}) \geq \gamma \alpha_k^2 \left| g(x_k)^T s_k + \frac{1}{2} d_k^T H(x_k) d_k \right|.$$

By (A.9) and (A.10),

$$\alpha_k^2 g(x_k)^T s_k + \frac{1}{2} \alpha_k^2 d_k^T H(x_k) d_k \rightarrow 0$$

for $k \rightarrow \infty, k \in \mathcal{L}$. Since both terms in this expression are nonpositive, it follows that

$$\alpha_k^2 g(x_k)^T s_k \rightarrow 0 \quad \text{and} \quad \alpha_k^2 d_k^T H(x_k) d_k \rightarrow 0.$$

Then either $\alpha_k \rightarrow 0$ or

$$\begin{cases} g(x_k)^T s_k \rightarrow 0, \\ d_k^T H(x_k) d_k \rightarrow 0 \end{cases}$$

for $k \rightarrow \infty, k \in \mathcal{L}$. In the first case we have $\alpha_k^2 \|s_k\| \rightarrow 0$ and $\alpha_k \|d_k\| \rightarrow 0$. In the second case, by Condition 1 we have $\alpha_k^2 \|s_k\| \rightarrow 0$. Moreover, Condition 1 also implies that $g(x_k) \rightarrow 0$ for $k \rightarrow \infty, k \in \mathcal{L}$, and hence, by using Condition 3, we obtain $\alpha_k \|d_k\| \rightarrow 0$ for $k \rightarrow \infty, k \in \mathcal{L}$. Therefore, by (A.2) we can conclude that

$$\lim_{k \rightarrow \infty} \alpha_k^2 \|s_k\| = 0 \quad \text{and} \quad \lim_{k \rightarrow \infty} \alpha_k \|d_k\| = 0,$$

which establishes (b). $\square$

Proof of Theorem 2.1. By Lemma A.1, the points x_k , $k = 0, 1, \dots$, remain in a compact set. If the algorithm terminates, the assertion is obvious. Therefore, let x_* be any limit point of $\{x_k\}$, and relabel $\{x_k\}$ a subsequence converging to x_* . By Lemma A.3, we have

$$(A.11) \quad \lim_{k \rightarrow \infty} \alpha_k^2 \|s_k\| = 0 \quad \text{and} \quad \lim_{k \rightarrow \infty} \alpha_k \|d_k\| = 0.$$

Thus, either

$$\lim_{k \rightarrow \infty} \|s_k\| = 0 \quad \text{and} \quad \lim_{k \rightarrow \infty} \|d_k\| = 0$$

or there exists a subsequence $\{x_k\}_{K_1}$ of $\{x_k\}$ such that $\alpha_k \rightarrow 0$ for $k \rightarrow \infty, k \in K_1$.

In the first case, taking into account the fact that s_k satisfies either Condition 1 or 2, we have $\lim_{k \rightarrow \infty} \|g(x_k)\| = 0$ and, by continuity, $g(x_*) = 0$; moreover, since $\lim_{k \rightarrow \infty} d_k^T H(x_k) d_k = 0$ by Condition 3 and by continuity

$$0 = \lim_{k \rightarrow \infty} \min [0, \lambda_{\min}(H(x_k))] = \min [0, \lambda_{\min}(H(x_*))].$$

In the second case, the point x_{k+1} , $k \in K_1$, is produced by the nonmonotone linesearch procedure, and hence there exists an index $\hat{k}$ such that, for all $k \geq \hat{k}$, $k \in K_1$,

$$\begin{aligned} f\left(x_k + \left(\frac{\alpha_k}{\sigma}\right)^2 s_k + \frac{\alpha_k}{\sigma} d_k\right) &> \hat{f}_k^R + \gamma \left(\frac{\alpha_k}{\sigma}\right)^2 \left[g(x_k)^T s_k + \frac{1}{2} d_k^T H(x_k) d_k\right] \\ &\geq f(x_k) + \gamma \left(\frac{\alpha_k}{\sigma}\right)^2 \left[g(x_k)^T s_k + \frac{1}{2} d_k^T H(x_k) d_k\right] \end{aligned}$$

for some $\sigma \in (0, 1)$. By the mean value theorem, we can find, for any $k \geq \hat{k}$, $k \in K_1$, a point $u_k = x_k + \omega_k \left((\alpha_k/\sigma)^2 s_k + (\alpha_k/\sigma) d_k\right)$ with $\omega_k \in (0, 1)$ such that, recalling that $g(x_k)^T d_k \leq 0$ we have

$$\begin{aligned} f\left(x_k + \left(\frac{\alpha_k}{\sigma}\right)^2 s_k + \frac{\alpha_k}{\sigma} d_k\right) &= f(x_k) + g(x_k)^T \left[\left(\frac{\alpha_k}{\sigma}\right)^2 s_k + \frac{\alpha_k}{\sigma} d_k\right] \\ &\quad + \frac{1}{2} \left[\left(\frac{\alpha_k}{\sigma}\right)^2 s_k + \frac{\alpha_k}{\sigma} d_k\right]^T H(u_k) \left[\left(\frac{\alpha_k}{\sigma}\right)^2 s_k + \frac{\alpha_k}{\sigma} d_k\right] \\ &\leq f(x_k) + \left(\frac{\alpha_k}{\sigma}\right)^2 g(x_k)^T s_k + \frac{1}{2} \left[\left(\frac{\alpha_k}{\sigma}\right)^2 s_k + \frac{\alpha_k}{\sigma} d_k\right]^T H(u_k) \left[\left(\frac{\alpha_k}{\sigma}\right)^2 s_k + \frac{\alpha_k}{\sigma} d_k\right]. \end{aligned}$$

It follows that

$$\begin{aligned} f(x_k) + \gamma \left(\frac{\alpha_k}{\sigma}\right)^2 \left[g(x_k)^T s_k + \frac{1}{2} d_k^T H(x_k) d_k\right] \\ \leq f(x_k) + \left(\frac{\alpha_k}{\sigma}\right)^2 g(x_k)^T s_k + \frac{1}{2} \left[\left(\frac{\alpha_k}{\sigma}\right)^2 s_k + \frac{\alpha_k}{\sigma} d_k\right]^T H(u_k) \left[\left(\frac{\alpha_k}{\sigma}\right)^2 s_k + \frac{\alpha_k}{\sigma} d_k\right]. \end{aligned}$$

Dividing both sides by $(\alpha_k/\sigma)^2$ and by simple manipulations we obtain

$$\begin{aligned} &(\gamma - 1) \left[g(x_k)^T s_k + \frac{1}{2} d_k^T H(x_k) d_k\right] \\ \text{(A.12)} \quad &\leq \frac{1}{2} d_k^T [H(u_k) - H(x_k)] d_k + \frac{1}{2} \left(\frac{\alpha_k}{\sigma}\right)^2 s_k^T H(u_k) s_k + \frac{\alpha_k}{\sigma} d_k^T H(u_k) s_k, \end{aligned}$$

where $\gamma < 1/2$. Now let $\{x_k\}_{K_2} \subset \{x_k\}_{K_1}$ be a subsequence such that

$$\lim_{\substack{k \rightarrow \infty \\ k \in K_2}} x_k = x_*, \quad \lim_{\substack{k \rightarrow \infty \\ k \in K_2}} s_k = s_*, \quad \lim_{\substack{k \rightarrow \infty \\ k \in K_2}} d_k = d_*.$$

From $\alpha_k \rightarrow 0$ for $k \rightarrow \infty$, $k \in K_1$, we have $u_k \rightarrow x_*$ as $k \rightarrow \infty$, $k \in K_2$. Since $\gamma - 1 < 0$, $g(x_k)^T s_k \leq 0$ and $d_k^T H(x_k) d_k \leq 0$ for all $k \in K_2$, then by (A.11) and (A.12) we have

$$\lim_{\substack{k \rightarrow \infty \\ k \in K_2}} (\gamma - 1) \left[g(x_k)^T s_k + \frac{1}{2} d_k^T H(x_k) d_k\right] = 0$$

and hence

$$g(x_*)^T s_* = 0 \quad \text{and} \quad d_*^T H(x_*) d_* = 0.$$

Condition 1 now implies that $g(x_*) = 0$, and Condition 3 gives that $H(x_*)$ is positive semidefinite. Moreover, by Lemma A.1 and Lemma A.3, we have that $x_* \in \Omega_0$. The

proof of assertion (a) can be easily completed similarly to the proof of Theorem 1 of [13].

Assertion (b) follows from Theorem 14.1.5 of [20] and by using Lemma A.3(b). Finally, assertion (c) follows from known results [20, p. 478], taking into account Lemma A.3(b). $\square$

Acknowledgments. We thank M. A. Saunders for providing us with the code of the SYMMLQ routine and J. K. Cullum for the complete code of Lanczos algorithms. Moreover, we are indebted to N. I. M. Gould and Ph. L. Toint for helpful discussions and for their constructive comments. Finally, we would like to thank very much the anonymous referee and A. Sartenaer (the second referee, who agreed to reveal her identity for this acknowledgment) for their useful suggestions and remarks which led to significantly improving the paper.

REFERENCES

- [1] M. ARIOLI, T. CHAN, I. DUFF, N. GOULD, AND J. K. REID, *Computing a Search Direction for Large-Scale Linearly-Constrained Nonlinear Optimization Calculations*, Technical report TR/PA/93/34, CERFACS, 1993.
- [2] I. BONGARTZ, A. CONN, N. GOULD, AND P. TOINT, *CUTE: Constrained and unconstrained testing environment*, ACM Trans. Math. Software, 21 (1995), pp. 123–160.
- [3] A. CONN, N. GOULD, AND P. TOINT, *LANCELOT: A Fortran Package for Large-Scale Nonlinear Optimization (Release A)*, Springer-Verlag, Heidelberg, Berlin, 1992.
- [4] J. CULLUM AND R. WILLOUGHBY, *Lanczos Algorithms for Large Symmetric Eigenvalue Computations*, Birkhauser, Boston, 1985.
- [5] R. S. DEMBO, S. C. EISENSTAT, AND T. STEihaug, *Inexact Newton methods*, SIAM J. Numer. Anal., 19 (1982), pp. 400–408.
- [6] R. DEMBO AND T. STEihaug, *Truncated-Newton methods algorithms for large-scale unconstrained optimization*, Math. Programming, 26 (1983), pp. 190–212.
- [7] N. DENG, Y. XIAO, AND F. ZHOU, *Nonmonotonic trust region algorithm*, J. Optim. Theory Appl., 76 (1993), pp. 259–285.
- [8] M. FERRIS, S. LUCIDI, AND M. ROMA, *Nonmonotone curvilinear linesearch methods for unconstrained optimization*, Comput. Optim. Appl., 6 (1996), pp. 117–136.
- [9] R. FLETCHER, *Practical Methods of Optimization*, John Wiley and Sons, New York, 1987.
- [10] G. GOLUB AND C. VAN LOAN, *Matrix Computations*, 2nd ed., The John Hopkins Press, Baltimore, 1989.
- [11] L. GRIPPO, F. LAMPARIELLO, AND S. LUCIDI, *A nonmonotone line search technique for Newton's method*, SIAM J. Numer. Anal., 23 (1986), pp. 707–716.
- [12] L. GRIPPO, F. LAMPARIELLO, AND S. LUCIDI, *A truncated Newton method with nonmonotone linesearch for unconstrained optimization*, J. Optim. Theory Appl., 60 (1989), pp. 401–419.
- [13] L. GRIPPO, F. LAMPARIELLO, AND S. LUCIDI, *A class of nonmonotone stabilization methods in unconstrained optimization*, Numer. Math., 59 (1991), pp. 779–805.
- [14] S. LUCIDI, F. ROCHETICH, AND M. ROMA, *Curvilinear Stabilization Techniques for Truncated Newton Methods in Large Scale Unconstrained Optimization: The Complete Results*, Technical report 02.95, Dipartimento di Informatica e Sistemistica, Università di Roma “La Sapienza,” 1995.
- [15] G. MCCORMICK, *A modification of Armijo's step-size rule for negative curvature*, Math. Programming, 13 (1977), pp. 111–115.
- [16] G. MCCORMICK, *Nonlinear Programming: Theory, Algorithm and Application*, John Wiley and Sons, New York, 1983.
- [17] J. MORÉ AND D. SORENSSEN, *On the use of directions of negative curvature in a modified Newton method*, Math. Programming, 16 (1979), pp. 1–20.
- [18] S. G. NASH, *Newton-type minimization via the Lanczos method*, SIAM J. Numer. Anal., 21 (1984), pp. 770–788.
- [19] S. G. NASH, *Preconditioning of truncated-Newton methods*, SIAM J. Sci. Stat. Comput., 6 (1985), pp. 599–616.

- [20] J. ORTEGA AND W. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.
- [21] C. C. PAIGE AND M. A. SAUNDERS, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.
- [22] B. PARLETT, *The Symmetric Eigenvalue Problem*, Prentice-Hall Series in Computational Mathematics, Prentice-Hall, Englewood Cliffs, NJ, 1980.
- [23] T. SCHLICK AND A. FOGELSON, *TNPACK—A truncated Newton package for large-scale problems: I. Algorithm and usage*, ACM Trans. Math. Software, 18 (1992), pp. 46–70.
- [24] G. A. SHULTZ, R. B. SCHNABEL, AND R. H. BYRD, *A family of trust-region-based algorithms for unconstrained minimization with strong global convergence properties*, SIAM J. Numer. Anal., 22 (1985), pp. 47–67.
- [25] D. C. SORENSEN, *Implicit application of polynomial filters in a k -step Arnoldi method*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 357–385.
- [26] P. TOINT, *Towards an efficient sparsity exploiting Newton method for minimization*, in *Sparse Matrices and Their Uses*, I. S. Duff, ed., Academic Press, London, 1981, pp. 57–88.
- [27] P. L. TOINT, *An assessment of nonmonotone linesearch techniques for unconstrained optimization*, SIAM J. Sci. Comput., 17 (1996), pp. 725–739.
- [28] P. TOINT, *Non-monotone trust-region algorithms for nonlinear optimization subject to convex constraints*, Math. Programming, 77 (1997), pp. 69–94.
- [29] Y. XIAO AND F. ZHOU, *Nonmonotone trust region methods with curvilinear path in unconstrained minimization*, Computing, 48 (1992), pp. 303–317.

NONDEGENERACY AND QUANTITATIVE STABILITY OF PARAMETERIZED OPTIMIZATION PROBLEMS WITH MULTIPLE SOLUTIONS*

J. FRÉDÉRIC BONNANS[†] AND ALEXANDER SHAPIRO[‡]

Abstract. This paper presents some results on quantitative stability of optimal solutions of parameterized optimization problems having nonisolated optima. The analysis is based on the introduced concept of nondegeneracy, which may be of independent interest. Examples of nonlinear and semidefinite programming are discussed.

Key words. sensitivity analysis, parameterized optimization, quantitative stability, semidefinite programming, nondegeneracy

AMS subject classifications. 90C30, 90C31

PII. S1052623497316518

1. Introduction. Consider the following parameterized optimization problem:

$$(1.1) \quad \min_{x \in X} f(x, u) \quad \text{subject to (s.t.) } x \in \Phi(u), \quad (P_u)$$

depending on the parameter vector $u \in U$. We assume that X is a finite dimensional vector space, say $X = \mathbb{R}^m$, that U is a normed vector space and that the feasible set is defined in the form

$$(1.2) \quad \Phi(u) = \{x : G(x, u) \in K\},$$

where K is a closed convex subset of a space $Y := \mathbb{R}^n$ and $G : X \times U \rightarrow Y$ is a *continuously differentiable* mapping. We investigate continuity properties of the set $\mathcal{S}(u)$ of optimal solutions of (P_u) , considered as a function of u .

We say that the optimal-solutions set $\mathcal{S}(u)$ is *Lipschitz stable* at a point $u_0 \in U$ if the multifunction $u \rightarrow \mathcal{S}(u)$ is upper Lipschitz continuous at u_0 ; i.e., there exist a constant $\alpha > 0$ and a neighborhood $\mathcal{U}$ of u_0 such that, for all $u \in \mathcal{U}$, the inclusion

$$(1.3) \quad \mathcal{S}(u) \subset \mathcal{S}(u_0) + \alpha \|u - u_0\| B_X$$

holds. (Here B_X denotes the unit ball $B_X := \{x \in X : \|x\| \leq 1\}$ in the space X .) In this paper we study Lipschitzian stability of optimal solutions in cases where $\mathcal{S}(u_0)$ possibly is not a singleton. In situations where the feasible set $\Phi(u) \equiv \Phi$ is fixed (independent of u), Lipschitzian stability of optimal solutions holds essentially under a second order growth condition (for the unperturbed problem); see [14]. The general case is considerably more subtle. If $\mathcal{S}(u_0) = \{x_0\}$ is a singleton, then Lipschitzian stability of optimal solutions was studied extensively (see, e.g., [5] and references

*Received by the editors February 12, 1997; accepted for publication (in revised form) November 11, 1997; published electronically August 26, 1998.

<http://www.siam.org/journals/siopt/8-4/31651.html>

[†]INRIA-Rocquencourt, Domaine de Voluceau, B.P. 105, 78153 Rocquencourt, France (frederic.bonnans@inria.fr).

[‡]School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0205 (ashapiro@isye.gatech.edu). The work of this author was supported in part by National Science Foundation grant DMI-9713878.

therein). It is known that the second order growth condition does not imply Lipschitzian stability of optimal solutions, even if $\mathcal{S}(u)$ is a singleton and the set K is polyhedral.

Therefore, in order to ensure Lipschitzian stability of optimal solutions, some additional assumptions are needed apart from the second order growth condition. In the next section we discuss a concept of nondegenerate points. We show that if a feasible point of the unperturbed problem is nondegenerate, then locally (P_u) can be reparameterized in such a way that the feasible set of the new problem does not depend on u . That is, in the nondegenerate case the question of Lipschitzian stability of optimal solutions can be reduced to studying parameterized problems with a fixed feasible set.

For the unperturbed problem, i.e., for $u = u_0$, we sometimes delete u_0 and write $G(\cdot)$ for $G(\cdot, u_0)$, etc. The Lagrangian, for the unperturbed problem (P) , can be written as

$$L(x, \lambda) := f(x) + \lambda \cdot G(x).$$

With an optimal solution x_0 of (P) is associated a set of Lagrange multipliers λ satisfying the first order optimality conditions:

$$(1.4) \quad D_x L(x_0, \lambda) = 0, \quad x_0 \in \Phi, \quad \lambda \in N_K(G(x_0)).$$

Here $\Phi := \Phi(u_0)$ is the feasible set of (P) , $N_K(y_0)$ denotes the normal cone to K at $y_0 \in K$, and $D_x L(x_0, \lambda)$ stands for the differential of $L(\cdot, \lambda)$ at the point x_0 . Under the constraint qualification, due to Robinson [11],

$$(1.5) \quad 0 \in \text{int}\{G(x_0) + DG(x_0)X - K\},$$

the set of Lagrange multipliers, satisfying (1.4), is nonempty and bounded [9, 10]. Note that since the space Y is finite dimensional, (1.5) is equivalent to

$$(1.6) \quad DG(x_0)X - T_K(G(x_0)) = Y,$$

where $T_K(y_0)$ denotes the tangent cone to K at $y_0 \in K$.

2. Nondegeneracy. In this section we discuss a concept of nondegeneracy for the unperturbed problem

$$(2.1) \quad \min_{x \in X} f(x) \quad \text{s.t.} \quad G(x) \in K. \quad (P)$$

The material of this section may be of independent interest and will be a basis for a stability analysis of the optimal solutions of (P_u) .

Consider a feasible point $x_0 \in \Phi$. Informally speaking we say that x_0 is a *nondegenerate* point of $G(x)$, with respect to the set K , if a local system of coordinates $\xi_1(y), \dots, \xi_k(y)$, $k \leq n$, can be constructed in Y such that the set K can be described near $y_0 := G(x_0)$ in that system of coordinates and the composite mapping $x \rightarrow (\xi_1(G(x)), \dots, \xi_k(G(x)))$ is nondegenerate at x_0 . The precise definition follows.

DEFINITION 2.1. *We say that a feasible point x_0 is a nondegenerate point of $G(x)$, with respect to K , if there exist (a) a neighborhood $\mathcal{N} \subset Y$ of $y_0 := G(x_0)$, (b) a continuously differentiable mapping $\Xi = (\xi_1, \dots, \xi_k) : \mathcal{N} \rightarrow \mathbb{R}^k$, and (c) a set $C \subset \mathbb{R}^k$ such that (i) $K \cap \mathcal{N} = \Xi^{-1}(C)$ and (ii) $D(\Xi \circ G)(x_0) : X \rightarrow \mathbb{R}^k$ is onto. We can assume, without loss of generality, that $\Xi(y_0) = 0$.*

Note that the above concept of nondegeneracy involves the mapping G and the set K and also depends on a particular choice of the local system of coordinates $\xi_1(y), \dots, \xi_k(y)$. The following simple example demonstrates this point.

Example 1. Consider the set $K := \{(y_1, y_2) : y_1 \geq 0\} \subset \mathbb{R}^2$ and the mapping $G(x) := (x, 0)$ from $\mathbb{R}$ into $\mathbb{R}^2$. Clearly then $\Phi = \{x : x \geq 0\}$. Consider the standard system of coordinates in $\mathbb{R}^2$, namely $\xi_1(y_1, y_2) := y_1$ and $\xi_2(y_1, y_2) := y_2$. We have then that $\Xi \circ G$ is linear, mapping x into $(x, 0)$, and hence, for any $x_0 \in \mathbb{R}$, $D(\Xi \circ G)(x_0) = \Xi \circ G$ is *not* onto. On the other hand the set K is defined by the constraint $y_1 \geq 0$, and hence we can take $\xi_1(y_1, y_2) := y_1$ with $k = 1$. For such choice of coordinates, $\Xi \circ G(x) = x$ and the nondegeneracy follows. Of course in the first representation the coordinate y_2 is redundant and the one-dimensional representation is preferable.

It follows from condition (i) of the above definition that locally the feasible set Φ can be defined by the constraint $(\Xi \circ G)(x) \in C$, i.e., $G^{-1}(K \cap \mathcal{N}) = (\Xi \circ G)^{-1}(C)$. Note that by the chain rule $D(\Xi \circ G) = D\Xi \circ DG$, and hence it follows from (ii) that $D\Xi(y_0) : Y \rightarrow \mathbb{R}^k$ is onto; i.e., the gradient vectors $\nabla \xi_1(y_0), \dots, \nabla \xi_k(y_0)$ are linearly independent, which is a required condition for $\xi_1, \dots, \xi_k$ to be a local system of coordinates. In fact it is not difficult to see that condition (ii) is equivalent to the following condition:

$$(2.2) \quad D\Xi(y_0)Y = \mathbb{R}^k \quad \text{and} \quad DG(x_0)X + \text{Ker}[D\Xi(y_0)] = Y.$$

Suppose that the point x_0 is nondegenerate, and consider the set

$$(2.3) \quad W := \{y \in \mathcal{N} : \xi_i(y) = 0, i = 1, \dots, k\}.$$

We have, by the condition (i) and since $\Xi(y_0) = 0$, that $W \subset K$ and, because of the first part of condition (2.2), W is a smooth manifold near y_0 , of dimension $n - k$, with the tangent space $T_W(y_0) = \text{Ker}[D\Xi(y_0)]$. Therefore, the second part of condition (2.2) can be written in the form

$$(2.4) \quad DG(x_0)X + T_W(y_0) = Y.$$

The above condition means that G intersects W transversally at the point x_0 (see, e.g., [6] for a discussion of the transversality concept).

Since $T_W(y_0) \subset T_K(y_0)$, it also follows from (2.4) that Robinson's constraint qualification (1.6) holds at x_0 ; i.e., the nondegeneracy condition is stronger than Robinson's constraint qualification. Therefore, if x_0 is a locally optimal solution of (P) , then there exists a Lagrange multiplier λ satisfying (1.4). Moreover, it follows from the nondegeneracy condition that λ is *unique*. Indeed, since $\lambda \in N_K(y_0)$ and $T_W(y_0) \subset T_K(y_0)$, we have that λ is orthogonal to $T_W(y_0)$. If λ' is another Lagrange multiplier, then $\lambda - \lambda'$ is orthogonal to both $DG(x_0)X$ and to $T_W(y_0)$. It follows then from (2.4) that $\lambda - \lambda' = 0$.

Let us note at this point that the above concept of nondegeneracy is somewhat different from the one introduced in [12], although it is aimed at essentially the same goal as the following result shows.

THEOREM 2.2. *Suppose that x_0 is a nondegenerate point of G . Then there exist neighborhoods $\mathcal{X}$ of x_0 , $\mathcal{U}$ of u_0 , $\mathcal{Z}$ of $0 \in \mathbb{R}^m$, and a continuously differentiable mapping $T : \mathcal{Z} \times \mathcal{U} \rightarrow \mathcal{X}$ such that $T(0, u_0) = x_0$ and, for every $u \in \mathcal{U}$, $T(\cdot, u)$ is a diffeomorphism of $\mathcal{Z}$ onto $\mathcal{X}$ (i.e., $T(\cdot, u) : \mathcal{Z} \rightarrow \mathcal{X}$ is one-to-one, onto, and its inverse is also continuously differentiable) and*

$$(2.5) \quad T((C \times \mathbb{R}^{m-k}) \cap \mathcal{Z}, u) = \Phi(u) \cap \mathcal{X}.$$

Proof. Consider the mapping $\Psi(x, u) := (\Xi \circ G)(x, u)$ defined on a neighborhood of (x_0, u_0) . By condition (ii) of Definition 2.1 we have that $D_x \Psi(x_0, u_0)$ is onto, or equivalently, that the $m \times k$ Jacobian matrix $\nabla_x \Psi(x_0, u_0)$ is of rank k . Without loss of generality we can assume that the upper $k \times k$ submatrix of $\nabla_x \Psi(x_0, u_0)$ is nonsingular. Let $\psi_i(x, u)$, $i = 1, \dots, k$, be coordinate functions of the mapping $\Psi(x, u)$, and consider the following system of equations:

$$(2.6) \quad z_1 = \psi_1(x, u), \dots, z_k = \psi_k(x, u), z_{k+1} = x_{k+1}, \dots, z_m = x_m.$$

We have then that the Jacobian, with respect to x , of this system at (x_0, u_0) is nonsingular, and hence, by the implicit function theorem, the above equations can be locally inverted in x for all u near u_0 . That is, for every u near u_0 , there exists a local diffeomorphism $T(\cdot, u) : \mathcal{Z} \rightarrow \mathcal{X}$ such that $z_i = \psi_i(T(z, u), u)$, $i = 1, \dots, k$, and $T(z, u)$ is continuously differentiable. Since locally the set $\Phi(u)$ can be defined by the constraint $(\psi_1(x, u), \dots, \psi_k(x, u)) \in C$, (2.5) follows. $\square$

The above theorem shows that *locally*, near x_0 and for all u in a neighborhood of u_0 , problem (P_u) can be reparameterized into the equivalent problem

$$(2.7) \quad \min_{z \in \mathbb{R}^m} f(T(z, u), u) \text{ s.t. } (z_1, \dots, z_k) \in C,$$

whose feasible set does not depend on u . We will use this in the next section while deriving some stability results for the optimal-solutions set $\mathcal{S}(u)$. Let us consider now some examples.

Example 2. Suppose that the set K can be defined near y_0 by a finite number of inequality constraints. That is, there exist continuously differentiable functions $\xi_1(y), \dots, \xi_k(y)$ and a neighborhood $\mathcal{N}$ of y_0 such that $\xi_i(y_0) = 0$, $i = 1, \dots, k$, and

$$(2.8) \quad K \cap \mathcal{N} = \{y \in \mathcal{N} : \xi_1(y) \geq 0, \dots, \xi_k(y) \geq 0\}.$$

In that case the set C is given by $\mathbb{R}_+^k$ and, in that system of coordinates, a feasible point x_0 is nondegenerate iff $\nabla \xi_1(y_0), \dots, \nabla \xi_k(y_0)$ are linearly independent and condition (2.4) holds with

$$T_W(y_0) = \{\eta : \eta^T \nabla \xi_i(y) = 0, i = 1, \dots, k\}.$$

Example 3. Suppose that the set K is polyhedral, and consider the tangent cone $T_K(y_0)$. Note that, since K is polyhedral, there is a neighborhood $\mathcal{N}$ of y_0 such that K coincides with $y_0 + T_K(y_0)$ in that neighborhood. Let $L := \text{lin}[T_K(y_0)]$ be the lineality space of $T_K(y_0)$, i.e., the largest linear subspace of $T_K(y_0)$. Then the cone $T_K(y_0)$ can be defined by a finite number of linear constraints $T_K(y_0) = \{y : a_i^T y \geq 0, i = 1, \dots, p\}$, with $a_i \in M$, where M is the linear space orthogonal to L and such that $L + M = Y$. Let $k := \dim(M)$ and choose a basis $b_1, \dots, b_k$ in M . We have then that $b_1, \dots, b_k$ are linearly independent and every vector a_i , $i = 1, \dots, p$, can be represented as a linear combination of vectors $b_1, \dots, b_k$. It follows that locally the set K can be defined in the following system of coordinates: $\xi_i(y) := b_i^T(y - y_0)$, $i = 1, \dots, k$. In that system of coordinates x_0 is nondegenerate iff

$$(2.9) \quad DG(x_0)X + \text{lin}[T_K(y_0)] = Y.$$

For *polyhedral* sets the above condition of nondegeneracy coincides with the one used in [12]. In particular if $K := \mathbb{R}_+^n$, then x_0 is nondegenerate iff vectors $\nabla g_i(x_0)$, $i \in I(x_0)$, are linearly independent, where $I(x_0) := \{i : g_i(x_0) = 0, i = 1, \dots, n\}$.

Finally let us consider the following example of semidefinite programming.

Example 4. Let $Y := S^{p \times p}$ be the space of $p \times p$ symmetric matrices and $K := S_+^{p \times p}$ be the cone of $p \times p$ positive semidefinite symmetric matrices. Let $y_0 \in S_+^{p \times p}$ be a matrix of rank $r < p$. Denote by $\lambda_1(A) \geq \dots \geq \lambda_p(A)$ the eigenvalues of a $p \times p$ symmetric matrix A and by $e_1(A), \dots, e_p(A)$ an orthonormal set of corresponding eigenvectors. Let $E(A)$ be the $p \times (p-r)$ matrix whose columns are formed from vectors $e_{r+1}(A), \dots, e_p(A)$, and let $E_0 := E(y_0)$. Note that the columns $e_{r+1}(y_0), \dots, e_p(y_0)$ of E_0 generate the null space of y_0 . Since $E(A)^T A E(A)$ is a diagonal matrix with diagonal elements $\lambda_{r+1}(A), \dots, \lambda_p(A)$ and the eigenvalues $\lambda_1(y_0), \dots, \lambda_r(y_0)$ are positive, we have that the cone $S_+^{p \times p}$ can be defined in a neighborhood of y_0 by the constraint $E(A)^T A E(A) \succeq 0$. (The notation $B \succeq 0$ means that the matrix B is positive semidefinite.) The latter constraint has value in the set of $(p-r) \times (p-r)$ symmetric matrices. However, this formulation is not suitable for our purpose since the eigenvectors $e_{r+1}(A), \dots, e_p(A)$ are not uniquely defined and $E(A)$ is not a continuous (and hence is not smooth) function of A near y_0 unless 0 is a simple eigenvalue of y_0 . In order to overcome this difficulty we proceed as follows.

Denote by $L(A)$ the eigenspace corresponding to the $p-r$ smallest eigenvalues of A , and let $P(A)$ be the orthogonal projection matrix onto $L(A)$. Also let E_0 be a (fixed) $p \times (p-r)$ matrix whose columns are orthonormal and span the space $L(y_0)$. It is well known (see [7]) that $P(A)$ is a continuously differentiable (in fact even analytic) function of A in a sufficiently small neighborhood of y_0 . Consequently $F(A) := P(A)E_0$ is also a continuously differentiable function of A in a neighborhood of y_0 , and, moreover, $F(y_0) = E_0$. It follows that, for all A sufficiently close to y_0 , the rank of $F(A)$ is $p-r$; i.e., its column vectors are linearly independent. Let $U(A)$ be the matrix whose columns are obtained by applying the Gram-Schmidt orthonormalization procedure to the columns of $F(A)$. The matrix $U(A)$ is well defined and continuously differentiable near y_0 , and, moreover, satisfies the following conditions: $U(y_0) = E_0$, the column space of $U(A)$ coincides with the column space of $E(A)$, and $U(A)^T U(A) = I_{p-r}$ (cf. [15, p. 557]).

We obtain that in a neighborhood of y_0 , the cone $S_+^{p \times p}$ can be defined in the form

$$(2.10) \quad \{A \in S^{p \times p} : U(A)^T A U(A) \succeq 0\}.$$

Consider the mapping $\Xi : A \rightarrow U(A)^T A U(A)$ from $\mathcal{N}$ into $S^{(p-r) \times (p-r)}$. The mapping Ξ is continuously differentiable, and, since

$$D\Xi(y_0)A = (DU(y_0)^T A)y_0 U(y_0) + U(y_0)^T A U(y_0) + U(y_0)^T y_0 (DU(y_0)A)$$

and $y_0 U(y_0) = 0$, we have $D\Xi(y_0)A = E_0^T A E_0$. It follows that $D\Xi(y_0)$ is onto. The set $W := \{A \in \mathcal{N} : \Xi(A) = 0\}$ is formed here by symmetric matrices of rank r , it is a smooth manifold and

$$(2.11) \quad T_W(y_0) = \{Z \in S^{p \times p} : E_0^T Z E_0 = 0\}.$$

We obtain that, in the above system of local coordinates, a feasible point x_0 is non-degenerate iff

$$(2.12) \quad DG(x_0)X + T_W(y_0) = S^{p \times p}.$$

Condition (2.12) means that G intersects *transversally*, at x_0 , the smooth manifold of symmetric matrices of rank r and is the standard nondegeneracy condition in semidefinite programming [1], [15, Section 2] (see also [16] for a discussion and equivalent formulations of this condition).

3. Stability analysis. Suppose that, for $u = u_0$, the corresponding set $S_0 := \mathcal{S}(u_0)$, of optimal solutions of the unperturbed problem (P) , is nonempty. In this section we investigate Lipschitzian stability of optimal solutions in cases where S_0 possibly is not a singleton.

In situations where the feasible set $\Phi(u) \equiv \Phi$ is fixed, it is relatively easy to give sufficient conditions for Lipschitzian stability of $\mathcal{S}(u)$. We say that the *second order growth* condition holds for (P) if there exist a constant $c > 0$ and a neighborhood $\mathcal{N}$ of S_0 such that

$$(3.1) \quad f(x) \geq f_0 + c [\text{dist}(x, S_0)]^2 \quad \forall x \in \Phi \cap \mathcal{N},$$

where $f_0 := \inf_{x \in \Phi} f(x)$ (see [2, 3, 4] for a discussion of this condition). Consider the optimization problem

$$(3.2) \quad \min_{x \in \Phi} \psi(x).$$

We can view $\psi(x)$ as a perturbation of the objective function $f(x)$ of the problem (P) . It is said that $\bar{x}$ is an ε -optimal solution of (3.2) if $\bar{x} \in \Phi$ and $\psi(\bar{x}) \leq \inf_{x \in \Phi} \psi(x) + \varepsilon$. We have then a simple bound for perturbations of an ε -optimal solution $\bar{x}$ in terms of the constant c and the Lipschitzian constant of the difference function $\psi(x) - f(x)$ [14, Lemma 2.1].

PROPOSITION 3.1. *Suppose that the second order growth condition (3.1) holds and that the difference function $\psi(x) - f(x)$ is Lipschitz continuous on the neighborhood $\mathcal{N}$ of S_0 , modulus κ , and let $\bar{x} \in \mathcal{N}$ be an ε -optimal solution of (3.2). Then*

$$(3.3) \quad \text{dist}(\bar{x}, S_0) \leq c^{-1}\kappa + c^{-1/2}\varepsilon^{1/2}.$$

We can formulate now the main result of this section. Recall that a function (a mapping) $g(\cdot)$ is said to be $C^{1,1}$ if it is continuously differentiable and its gradient $\nabla g(\cdot)$ is locally Lipschitz continuous.

THEOREM 3.2. *Suppose that (i) $f(x, u)$ and $G(x, u)$ are $C^{1,1}$, (ii) the second order growth condition (3.1) (for the unperturbed problem) holds, (iii) every point of the set S_0 is a nondegenerate point of G , with respect to K , and with the corresponding coordinate functions $\xi_i(y)$ being $C^{1,1}$ functions, and (iv) for all u in a neighborhood of u_0 , the optimal-solutions sets $\mathcal{S}(u)$ are uniformly bounded. Then $\mathcal{S}(u)$ is Lipschitz stable at u_0 .*

Proof. Consider a sequence $u_n \rightarrow u_0$, and let $x_n \in \mathcal{S}(u_n)$. It suffices to show that

$$(3.4) \quad \text{dist}(x_n, S_0) = O(\|u_n - u_0\|).$$

Since $\mathcal{S}(u)$ are uniformly bounded, we can assume, by passing to a subsequence if necessary, that the sequence $\{x_n\}$ converges to a point x_0 . By continuity arguments it is not difficult to show (and it is well known) that then $x_0 \in S_0$. Since it is assumed that the point x_0 is nondegenerate, by Theorem 2.2 there exists a continuously differentiable mapping $T(z, u)$ such that, locally near x_0 and for all u in a neighborhood of u_0 , problem (P_u) can be reparameterized into the problem (2.7). Note that, for n large enough, the relation $T(z_n, u_n) = x_n$ defines a sequence $z_n \rightarrow 0$ of optimal solutions of (2.7) for $u = u_n$. Let Z_0 be the set of optimal solutions of the reparameterized problem (2.7) restricted to a neighborhood of 0. Since $T(\cdot, u)$ is Lipschitz continuous near $z = 0$ with the Lipschitz constant uniformly bounded for all u in a neighborhood of u_0 , the estimate (3.4) will follow from

$$(3.5) \quad \text{dist}(z_n, Z_0) = O(\|u_n - u_0\|).$$

It remains to show that the Lipschitzian stability result (3.5), for the reparameterized problem, holds. First let us observe that since the mapping $T(\cdot, u_0)$ and its inverse are continuously differentiable, and hence are Lipschitz continuous near x_0 , we have that the second order growth condition for the problem (2.7), possibly restricted to a neighborhood of $z = 0$ and for $u = u_0$, follows from the second order growth condition (3.1). Therefore, by (3.3), we only have to verify that the Lipschitz constant of the function $h(\cdot, u) := f(T(\cdot, u), u) - f(T(\cdot, u_0), u_0)$, in a neighborhood $\mathcal{Z}$ of 0, is of order $O(\|u - u_0\|)$. Note that by the mean value theorem such a Lipschitz constant is given by $\sup_{z \in \mathcal{Z}} \|\nabla_z h(z, u)\|$. Therefore, by the chain rule of differentiation and since f is a $C^{1,1}$ function, we have that the Lipschitz constant of $h(\cdot, u)$, on $\mathcal{Z}$, is of order $O(\|u - u_0\|)$ if

$$(3.6) \quad \sup_{z \in \mathcal{Z}} \|\nabla_z T(z, u) - \nabla_z T(z, u_0)\| = O(\|u - u_0\|).$$

Since the coordinate mapping Ξ is assumed to be $C^{1,1}$, the composite mapping $\Xi \circ G$ is also $C^{1,1}$. The implicit function theorem implies then that the mapping $T(z, u)$ is also $C^{1,1}$. This completes the proof. $\square$

In the case of nonlinear programming, i.e., when the set K is polyhedral, the above stability result was derived in [13] (under somewhat restrictive assumptions) and in [8] by different techniques. This result is new in the framework of semidefinite programming. The stability result can be also extended to ε -optimal-solutions set of (P_u) if $\varepsilon = O(\|u - u_0\|^2)$.

REFERENCES

- [1] F. ALIZADEH, J.P. HAEBERLY, AND M.L. OVERTON, *Complementarity and nondegeneracy in semidefinite programming*, Math. Programming, 77 (1997), pp. 111–128.
- [2] J.F. BONNANS, *Extended quadratic tangent optimization problems*, in Mathematical Programming with Data Perturbations, A.V. Fiacco, ed., Marcel Dekker, New York, Basel, 1998, pp. 31–45.
- [3] J.F. BONNANS AND A.D. IOFFE, *Second-order sufficiency and quadratic growth for nonisolated minima*, Math. Oper. Res., 20 (1995), pp. 801–817.
- [4] J.F. BONNANS AND A.D. IOFFE, *Quadratic growth and stability in convex programming problems with multiple solutions*, J. Convex Anal., 2 (1995), pp. 41–57.
- [5] J.F. BONNANS AND A. SHAPIRO, *Optimization problems with perturbations: A guided tour*, SIAM Rev., 40 (1998), pp. 228–264.
- [6] M. GOLUBITSKY AND V. GUILLEMIN, *Stable Mappings and their Singularities*, Springer-Verlag, New York, 1973.
- [7] T. KATO, *Perturbation Theory for Linear Operators*, Springer-Verlag, New York, 1984.
- [8] D. KLATTE, *On quantitative stability for non-isolated minima*, Control Cybernet., 23 (1994), pp. 183–200.
- [9] S. KURCYUSZ, *On the existence and nonexistence of Lagrange multipliers in Banach spaces*, J. Optim. Theory Appl., 20 (1976), pp. 81–110.
- [10] S.M. ROBINSON, *First order conditions for general nonlinear optimization*, SIAM J. Appl. Math., 30 (1976), pp. 597–607.
- [11] S.M. ROBINSON, *Stability theory for systems of inequalities, part II: Differentiable nonlinear systems*, SIAM J. Numer. Anal., 13 (1976), pp. 497–513.
- [12] S.M. ROBINSON, *Local structure of feasible sets in nonlinear programming. II. Nondegeneracy*, Math. Programming Stud., 22 (1984), pp. 217–230.
- [13] A. SHAPIRO, *Perturbation theory of nonlinear programs when the set of optimal solutions is not a singleton*, Appl. Math. Optim., 18 (1988), pp. 215–229.
- [14] A. SHAPIRO, *Perturbation analysis of optimization problems in Banach spaces*, Numer. Funct. Anal. Optim., 13 (1992), pp. 97–116.
- [15] A. SHAPIRO AND M.K.H. FAN, *On eigenvalue optimization*, SIAM J. Optim., 5 (1995), pp. 552–569.
- [16] A. SHAPIRO, *First and second order analysis of nonlinear semidefinite programs. Semidefinite programming*, Math. Programming Ser. B, 77 (1997), pp. 301–320.

PROGRESS MADE IN SOLVING THE MULTICOMMODITY FLOW PROBLEM*

RICHARD D. MCBRIDE†

Abstract. During the past few years, steady progress has been made in solving the multicommodity flow problem using EMNET, a primal partitioning network simplex algorithm. This paper describes a recent improvement made to EMNET which permits it to dramatically reduce the solution time for multicommodity flow problems with large numbers of side constraints. It is now possible to efficiently solve extremely large multicommodity flow problems when this new idea is coupled with the resource-directive decomposition heuristic as found in EMNET. Commercial problems with the number of constraints ranging up to 600,000 are routinely solved by EMNET.

Key words. partitioned inverse, multicommodity, transportation

PII. S1052623496304542

1. Introduction. Many large-scale production, distribution planning, and management problems can be formulated as multicommodity flow problems. The classical formulation usually describes a distribution management model. This model describes the flows of products (commodities) through a distribution system as a collection of networks, one for each product. Side constraints are added to the model to limit the total amount of flow between certain origin/destination pairs. These joint capacity constraints describe inventory storage, loading dock, and/or truck (train) limitations. Surveys of the application of multicommodity flow problems have been published by Tomlin (1966), Kennington (1978), and Assad (1978).

Formulations of ever-increasing size and complexity are being proposed to capture economies of scope and coordination among plants and warehouses. Fortunately, as the models have increased in size, the cost of computation has dropped to a point where it is now feasible to solve very large multicommodity flow problems on hardware of modest cost. A few years ago research on large multicommodity flow problems was done on supercomputers (i.e., Cray-YMP) and other computers with multiple processors (e.g., Sequent-20 processors, KORBX-14 processors). Research is still continuing on computers with multiple processors. Lustig and Rothberg (1995) report work done on a Silicon Graphics Power Challenge with 16 processors. As will be seen in this paper, these expensive computers are not required to efficiently solve the multicommodity flow problem.

Because there is a separate network for each commodity, researchers have tried decomposition solution techniques. In this approach the network flow problem for each commodity is solved, then adjustments to either costs or capacities are made and the problems are reoptimized. The adjustments are made to move the decomposition solution toward a solution that satisfies the side constraints and is optimal. Decomposition algorithms have been proposed for the multicommodity flow problem by Zenios, Pinar, and Dembo (1995), Pinar and Zenios (1992), and Schultz and Meyer (1991).

During the last few years, steady progress has been made in using the simplex method to solve this class of problems. The multicommodity flow problem can be

*Received by the editors May 28, 1996; accepted for publication (in revised form) November 7, 1997; published electronically August 26, 1998.

<http://www.siam.org/journals/siopt/8-4/30454.html>

†School of Business, University of Southern California, Los Angeles, CA 90089-1421 (mcbride@rcf.usc.edu).

viewed as a network problem with side constraints. The special network structure can be exploited by using a primal basis partition. The key point is that only a network basis and a small portion of the basis inverse are needed to efficiently execute the simplex method. This small square submatrix of the basis inverse, the working inverse, has a dimension equal to the number of side constraints. The network basis naturally partitions by commodity. Using the efficient network technology that has been fine-tuned to handle the network portion of the problem, the primal partitioning network simplex approach becomes very competitive. EMNET, developed in McBride (1985), is an implementation of the primal partitioning network simplex approach. With the enhancements described in McBride and Mamer (1995a) and the subsequent work presented in this paper, a very efficient algorithm has been developed for the solution of the multicommodity flow problem on single-processor computers.

Castro and Nabona (1996) also present an implementation of a primal partitioning algorithm. The partitioning concepts of their approach are similar but their mathematical development and implementation are quite different to those of McBride (1985) and McBride and Mamer (1995a). They don't use the resource-directive decomposition heuristic presented in McBride and Mamer (1995a). They show good solution times but unfortunately don't use any of the KEN or PDS test problems to give us a more direct comparison with their algorithm.

Large multicommodity flow problems may have a substantial number of side constraints due to inventory, loading dock, and/or truck (train) capacities. This causes a rather large working inverse to be maintained and updated on each pivot. Surprisingly, for most of the large multicommodity flow problems considered in this paper, EMNET spends a large portion of its time in that part of the solution process which involves the working inverse. Even though this inverse can be represented using an LU factorization, significant work is still required when using the inverse on each step of the simplex method.

While the number of side constraints may be large for a good-sized multicommodity flow problem, the number of binding side constraints at optimality may be quite small. Unfortunately it is unknown which will be binding at optimality until the problem is solved. Working with just the binding constraints requires the ability to efficiently change the dimension of the working inverse from one pivot to the next. Castro and Nabona (1996) present a way to reduce the dimension of the LU representation. They project how big it can possibly get before it is refactored to make room for any possible increase in the number of variables in the basis, but present no computational experience on the gains made using their basis-reduction concepts.

In this paper we describe a heuristic approach to control the size of the working inverse. Using this approach we were able to reduce the size of the working inverse by more than 70% for problems PDS-30, PDS-40, PDS-50, PDS-60, PDS-70, and PDS-80. This reduction yielded significant solution-time reductions. A 40% reduction on the solution time for PDS-40 was obtained, while PDS-30, PDS-50, PDS-60, PDS-70, and PDS-80 had larger percentage reductions. The reduction was more pronounced for the larger problems. The two smaller problems PDS-10 and PDS-20 had 17% and 31% reduction in solution times, respectively. In section 3 we describe this heuristic approach, after describing EMNET in section 2.

Section 4 more fully describes our computational results with the KEN and PDS family of test problems. These two families of problems do not exhaust all possible multicommodity flow problem instances. They do, however, allow for comparison of our results with results published elsewhere, since they are a widely used suite of test

problems. EMNET is also successfully being used to solve large commercial multi-commodity flow problems on a regular basis with problem constraint sizes ranging up to 600,000.

2. EMNET. Both pure and generalized network substructures are exploited by EMNET. This algorithm has its origins in GENNET from Brown and McBride (1984). The GENNET program is a generalized network code that utilizes all of the modern efficient generalized network data structures. The GENNET algorithm was extended to allow two arbitrary entries in each column, side constraints, and side columns to produce EMNET as described in McBride (1985).

As mentioned above, the multicommodity flow problem can be viewed as a network problem with side constraints. Our problem can be written in matrix notation as

$$(1) \quad \begin{aligned} &\min cx, \\ &Gx = b, \\ &Ax = r, \\ &0 \leq x \leq u. \end{aligned}$$

Here G consists of a separate network for each commodity. A is the coefficient matrix of the side constraints with slack variables added.

In EMNET the basis is partitioned into a network part and a nonnetwork part. The basic network part is further partitioned into a network basis (nonsingular square submatrix) G_1 and the remaining part G_2 . Partitioning the basic part of A to conform to the $[G_1, G_2]$ partition yields $[A_1, A_2]$. The full basis partition is as follows:

$$B = \begin{bmatrix} G_1 & G_2 \\ A_1 & A_2 \end{bmatrix}.$$

The partitioned basis inverse is given by

$$B^{-1} = \begin{bmatrix} (I + G_1^{-1}G_2H^{-1}A_1)G_1^{-1} & -G_1^{-1}G_2H^{-1} \\ -H^{-1}A_1G_1^{-1} & H^{-1} \end{bmatrix},$$

where

$$(2) \quad H = A_2 - A_1G_1^{-1}G_2.$$

The duals are computed according to the standard formulas $c_B B^{-1}$ which, when partitioned as above, yields

$$\begin{aligned} w_2 &= (c_2 - c_1G_1^{-1}G_2)H^{-1}, \\ w_1 &= (c_1 - w_1A_1)G_1^{-1}, \end{aligned}$$

where the cost vector $c = [c_1, c_2]$ has been partitioned in a fashion conformable to G_1 and G_2 . The w_2 -values correspond to the dual variables for the side constraints and the w_1 -values to the dual variables for the nodes in the network. EMNET uses H^{-1} , G_1 , and the original problem data to execute the simplex method. The G_1 submatrix is maintained using the standard depth, predecessor, and traversal arrays put forward by Brown and McBride (1984) for the solution of generalized network problems. The

working inverse, H^{-1} , has dimension equal to the number of side constraints and is maintained in LU factorization form as described in McBride and Mamer (1995b). The w_1 -values are computed on demand during pricing.

EMNET has the ability to handle piecewise-linear convexity in the objective coefficients as described in McBride and Mamer (1995a). The separable piecewise-linear convexity is not only important in its own right but turns out to be very important in the implementation of the resource-directive decomposition heuristic.

The solution approach used by EMNET is to first drop the side constraints and solve the resulting network problems. The side constraints are added back and then EMNET proceeds to feasibility and optimality. CPLEX, OSL, and others use this same approach. Instead of using the partitioned inverse given above, they use an LU factorization of the full basis after the side constraints are added back. This is one of the main differences between their simplex approach and EMNET. Because of the size of the full basis, their LU factorization can have a considerable number of nonzero entries and therefore require a great amount of work to execute simplex pivots. Since the network solution may not be close to the optimal solution, a large number of pivots using the LU-factored basis may be required to achieve optimality. The network pivots are extremely fast but when the full basis or even H in LU-factored form is used the time per pivot greatly increases. To overcome this problem, a primal allocation heuristic was developed to move the network solution to one which is nearly feasible and at the same time much closer to the optimal solution. This is fully described in McBride and Mamer (1995a).

The primal allocation heuristic can be viewed as a resource-directive decomposition procedure. Working solely with the network part of the problem, we first solve the network problem(s) resulting from temporarily dropping the side constraints. The side constraints are then checked for violation. The positive variables in violated side constraints are proportionally assigned new upper bounds so that if each variable were reduced in value to its new upper bound, all side constraints would be satisfied. The new upper bounds are enforced by the use of higher objective costs imposed using the piecewise-separable convex capabilities of EMNET. The network problems are reoptimized. As the new bounds are imposed, variables that were previously zero may take on positive values. A new proportional assignment of upper bounds is made across all positive variables and the networks are again reoptimized. This process continues until there is no further improvement made to the total infeasibility of the side constraints. For the problems considered in this paper the total infeasibility of the side constraints is roughly cut in half on each iteration of this process, as seen in Table 1.

Table 1 shows the results for PDS-80. PDS-80 has 120,879 nodes, 8,302 side constraints, and 434,580 variables partitioned among 11 commodities. The heuristic took 213 seconds, which includes the initial network solution times on a Cobra 300. The optimal objective value is 11,469,077,462. When the process stopped, constraint 6,943 was the most violated constraint with an infeasibility of 0.00097. The total infeasibility of all the constraints was 0.0190. The first 2 digits of the objective function are correct and the third digit is off by 2 at the end of the heuristic.

The heuristic not only has worked well for the set of test problems considered in this paper but has also worked well on other problems encountered in practice. Unfortunately we do not have permission to discuss these additional results.

3. EMNET basis-reduction heuristic. In the implementation of EMNET discussed above, the dimension of H in (2) is equal to the number of side constraints.

TABLE 1
Resource-directive decomposition heuristic results for PDS-80.

Trial	Infeasibility	Objective value
0	211,103.	7,495,306,244.
1	92,359.	9,036,458,587.
2	46,977.	9,985,043,022.
3	19,189.	10,921,964,657.
4	7,316.	11,361,108,600.
5	2,797.	11,553,945,452.
6	687.	11,646,428,810.
7	206.	11,668,564,921.
8	103.	11,675,827,074.
9	10.9	11,678,949,601.
10	2.7	11,679,343,919.
11	0.0299	11,679,403,837.
12	0.0190	11,679,404,689.

When a side constraint is not binding, the slack variable in the constraint is basic. Even though each slack variable has only one nonzero coefficient in its column, the coefficients in the constraints for the other basic variables are in the basis. EMNET must also do computations on each pivot to track the feasibility of the nonbinding side constraints. This increases the work required to perform a pivot of the simplex method.

The EMNET basis-reduction heuristic is very simple conceptually yet yields very surprising results. The fundamental algorithmic steps of EMNET are now outlined to explain the heuristic and place it in perspective.

1. Drop the side constraints and solve the resulting network problem(s).
2. Apply the resource-directive decomposition heuristic to gain near feasibility of the side constraints.
3. Add the side constraints that are binding and that are within $T_1\%$ of being violated. Let h be the number of side constraints added. We call the added side constraints the active side constraints and the remaining side constraints the inactive side constraints. The dimension of the working inverse, H^{-1} , is h . EMNET proceeds with the active side constraints while ignoring the inactive side constraints.
4. As EMNET proceeds it moves constraints from the inactive set to the active set as the solution moves to violate the constraints. The dimension of H , h , increases as side constraints are added to the active set. This can be done by adding those inactive side constraints that become violated or within $T_2\%$ of being violated.

There are many ways to implement steps 3 and 4. Let m be the total number of side constraints. One implementation would be to map the active constraints so that they are renumbered $1, 2, \dots, h$ and the inactive constraints $h+1, h+2, \dots, m$. As each side constraint coefficient is encountered during the execution of the algorithm, its mapped constraint number is checked. If its mapped row number is greater than h , it is skipped.

In step 3 the violated constraints (usually they are only slightly violated) and those within T_1 relative percent of being violated are placed into the active set. This means that only those satisfied side constraints are added that have the smaller slack values. The greater satisfied side constraints are left in the inactive set. As EMNET

TABLE 2
Problems solved.

Problem	Nodes	Side const.	Total rows	Total columns	Commodities
KEN-11	14,641	53	14,694	21,346	121
KEN-13	28,561	71	28,632	42,659	169
KEN-18	104,976	151	154,127	154,699	324
PDS-10	15,389	1,169	16,558	49,932	11
PDS-20	31,427	2,447	33,874	108,175	11
PDS-30	46,453	3,491	49,944	158,489	11
PDS-40	62,172	4,672	66,844	217,531	11
PDS-50	77,341	5,719	83,060	275,814	11
PDS-60	92,653	6,778	99,431	336,421	11
PDS-70	107,250	7,694	114,944	390,005	11
PDS-80	120,879	8,302	129,181	434,580	11

proceeds and as constraints in the inactive set approach infeasibility or become infeasible, they are added to the active set. Since changing the dimension of an LU factorization of H on any one pivot is inefficient, it seems more reasonable to do it at the time of refactorization. Periodically a new LU factorization of H is recomputed to reduce the number of nonzero elements in its representation and control roundoff error. At this time the feasibility of the inactive side constraints is checked. Those inactive side constraints that are violated or within $T_2\%$ of being violated are placed into the active set. The dimension of H increases with the addition of constraints into the active set. The working inverse enters the refactorization process with dimension h_{old} and emerges with a slightly larger dimension h_{new} when constraints are added.

4. Computational experience. Computational experience is presented using the KEN and PDS sets of problems. These problems are not an exhaustive set of multicommodity flow test problems but are readily available to researchers and have been widely reported upon. Subsets of these problems have been reported on by Lustig and Rothberg (1995), Pinar and Zenios (1992), Zenios, Pinar, and Dembo (1995), Schultz and Meyer (1991), Marsten et al. (1990), and Carolan et al. (1990). Problems KEN-11, KEN-13, KEN-18, PDS-10, and PDS-20 are in the NETLIB archive. The full PDS family of problems was generated using software obtained from Dr. Robert Meyer. Table 2 gives the number of nodes, side constraints, variables (including slacks in the side constraints), and commodities for the two sets of problems.

All of these problems were solved to optimality using a Carrera Cobra 300 with 64 megabytes of RAM. The Carrera Cobra 300 has a single 21064A Alpha processor running at 300 MHZ. Table 3 gives the number of side constraints in each problem, the dimension of H at optimality when using the heuristic, the number binding at optimality, and the percent binding at optimality. T_1 was set equal to 0.15 for the PDS problems and 0.02 for the KEN problems. A value of T_2 equal to 0.1 was used for both sets of problems. In the PDS problem set the number of binding side constraints ranges from 24% to 28% of the number of side constraints. For each of the PDS problems the dimension of H is within 10% of its minimum value, the number of binding side constraints at optimality. This is encouraging because it shows that very little would be gained in further reducing the dimension of H to its minimum dimension.

The number of side constraints in the KEN problems is so small that the heuristic gives very small reductions in solution time. With such a large reduction in the size

TABLE 3
Reduction in size of working inverse.

Problem	Number of side constraints	Dim of H at optimality	Number binding at optimality	Binding
KEN-11	53	12	9	17%
KEN-13	71	31	17	24%
KEN-18	151	50	32	21%
PDS-10	1,169	338	311	27%
PDS-20	2,447	749	688	28%
PDS-30	3,491	968	898	26%
PDS-40	4,672	1,330	1,199	26%
PDS-50	5,719	1,617	1,502	26%
PDS-60	6,778	1,944	1,776	26%
PDS-70	7,694	2,079	1,928	25%
PDS-80	8,303	2,184	2,012	24%

TABLE 4
Improved solution times.

Problem	Solution time in sec. of full working inverse	Solution time in sec. using heuristic	Improvement
KEN-11	3	3	0%
KEN-13	19	18	5%
KEN-18	70	66	5%
PDS-10	12	10	17%
PDS-20	169	116	31%
PDS-30	548	325	41%
PDS-40	1,764	1,065	40%
PDS-50	2,488	1,366	45%
PDS-60	4,565	2,504	45%
PDS-70	6,393	3,607	44%
PDS-80	9,112	5,105	44%

of the working inverse for the PDS problems we would expect a significant reduction in the solution time. This is exactly the case as shown in Table 4. The solution times using the full working inverse are from McBride and Mamer (1995b). A 40% or more reduction in solution time for PDS-30, PDS-40, PDS-50, PDS-60, PDS-70, and PDS-80 was obtained. The four largest PDS problems have a 44% or more reduction in solution time. The basis-reduction heuristic yields a very significant reduction in solution times for the PDS problems with large numbers of side constraints.

Table 5 catalogs the known solution times (all known to the author) for the KEN and PDS problems. The solution methods used were decomposition methods with 1 and 8 processors on the Cray-YMP by Pinar and Zenios (1992) and Zenios, Pinar, and Dembo (1995). They were able to get approximately 5 digits of accuracy in both the objective and the feasibility of the side constraints. Marsten et al. (1990) used an interior point algorithm on a Cray-YMP. Schultz and Meyer (1991) used a decomposition method coupled with the interior point method on a 20-processor Sequent machine. The EMNET results imply they obtained 3 digits of accuracy in the optimal objective value for PDS-30, PDS-40, and PDS-50 and 2 digits of accuracy for PDS-60 and PDS-70. The OSL (Optimization System Library) results were obtained using the primal simplex method hot started with the network solution on an IBM

TABLE 5
Known results (times in seconds).

Prob.	Z1	Z8	M	S&M	KBX	OSL	L&R	EMNET
Ken-11	16		23		396	27		3
Ken-13	138		67		1,242	573		18
Ken-18			680		15,840	12,764		66
PDS-10	408	96	1,521	999	11,880	475		10
PDS-20	1,946	740	15,972	3,043	63,720	4,324	409	116
PDS-30	7,504	2,566		6,480		45,069		325
PDS-40				10,440				1,065
PDS-50				19,800				1,366
PDS-60				24,900				2,504
PDS-70				33,840				3,607
PDS-80								5,105
Z1:	Pinar & Zenios (1992) & Zenios, Pinar, and Dembo (1995), Cray-YMP, one processor. Approx. 5 digits of accuracy. Decomposition method is used.							
Z8:	Pinar & Zenios (1992) & Zenios, Pinar, and Dembo (1995), Cray-YMP, Eight processors. Approx. 5 digits of accuracy. Decomposition method is used.							
M:	Marsten et al. (1990), Cray-YMP, one processor. Interior point method is used.							
S&M	Schultz & Meyer (1991), Sequent 11 processors stopped after 50 iterations. Time reported is elapsed time. Three digits of accuracy for PDS-30 to 50, two digits of accuracy for PDS-60 & PDS-70. Decomposition method is used.							
KBX:	Carolán et al. (1990), KORBX, 14 processors. Time reported is elapsed time. Interior point method is used.							
OSL:	IBM ES9000 mainframe. Hot stated by solving pure network relax. with network code first. Simplex method is used.							
L&R:	Lustig & Rothberg (1995), SGI Power Challenge with 16 R8000 processors & 4 Gbytes of ram. Interior point method is used. The modified CPLEX code used 2404 sec. with one processor.							
EMNET	Carrera Cobra 300 with 300 MHZ 21064A Alpha processor and 64 mega-bytes of RAM. Embedded network primal simplex method is used.							

mainframe. Using the network solution as the starting basis, the full basis was used as the algorithm moved from the network starting basis to the optimal basis. See McBride and Mamer (1995a) for a further discussion of these results. These OSL results clearly show the difficulty the simplex method has when forced to deal with extremely large working inverses. The Lustig and Rothberg results were obtained on an SGI Power Challenge with 16 processors using a modified CPLEX interior point algorithm. EMNET, a primal partitioning network simplex implementation, uses 1 Alpha 21064A processor. Table 5 clearly shows the superiority of the EMNET approach using a single-processor machine.

5. Conclusions and implications. The basis-reduction heuristic dramatically reduces the solution times for the PDS problems. It appears that the basis-reduction heuristic coupled with the resource-directive decomposition heuristic provides a real breakthrough in the solution of large multicommodity flow problems.

It appears that the simplex method is the best way to solve large multicommodity flow problems when the network portion is isolated and exploited. Table 5 clearly shows the superiority of the primal partitioned network simplex approach over decom-

position and interior point methods. Expensive hardware with multiple processors is not needed to solve large multicommodity flow problems. Extremely fast solution times can be obtained on modest hardware.

Given that we can quickly solve the continuous multicommodity flow problem, we can now consider problems which have the continuous multicommodity flow problem as a relaxation. One of these is the sole-sourcing problem in its various forms. This simplex basis allows for easy reoptimization of the problems essential for solving the sole-sourcing problems. Sole sourcing involves shipping all of the products a customer orders from the same (sole) plant whenever possible. When the multicommodity flow problem is solved, customer orders can be split between plants. Some kind of postprocessing of the continuous solution (intelligently guided reoptimizations) is required to get sole-sourced solutions. Strictly speaking this is an integer programming problem, but when the problem can have millions of variables, the standard branch and bound procedures need to be modified.

EMNET has been successfully used to solve multicommodity flow problems requiring sole sourcing for clients but we do not have permission to discuss the work done. Even though we obtained excellent results we feel that there is much to be done in this area. Given that EMNET is able to quickly and efficiently solve the basic large multicommodity flow problem, it now puts the sole-sourcing and other related integer problems center stage for further research.

REFERENCES

- A. ASSAD (1978), *Multicommodity network flows: A survey*, Networks, 8, pp. 37–92.
- G. BROWN AND R. MCBRIDE (1984), *Solving generalized networks*, Management Sci., 30, pp. 1497–1523.
- W. CAROLAN, J. HILL, J. KENNINGTON, S. NIEMI, AND S. WICHMANN (1990), *An empirical evaluation of the KORBX algorithm for military airlift applications*, Oper. Res., 38, pp. 240–248.
- J. CASTRO AND N. NABONA (1996), *An implementation of linear and nonlinear multicommodity network flows*, European J. Oper. Res., 92, pp. 37–53.
- J. KENNINGTON (1978), *A survey of linear multicommodity network flows*, Oper. Res., 26, pp. 209–236.
- I. J. LUSTIG AND E. ROTHBERG (1995), *Gigaflows in Linear Programming*, working paper. (This paper is available from the authors.)
- R. MARSTEN, R. SUBRAMANIAN, I. LUSTIG, AND D. SHANNO (1990), *Interior point methods for linear programming: Just call Newton, Lagrange, and Fiacco and McCormick!*, Interfaces, 20, pp. 105–116.
- R. MCBRIDE (1985), *Solving embedded generalized network problems*, European J. Oper. Res., 21, pp. 82–92.
- R. MCBRIDE AND J. MAMER (1995), *Solving multicommodity flow problems*, working paper.
- R. D. MCBRIDE AND J. W. MAMER (1997), *Solving multicommodity flow problems with a primal embedded network simplex algorithm*, INFORMS J. Comput., 9, pp. 154–163.
- M. PINAR AND S. ZENIOS (1992), *Parallel decomposition of multicommodity network flows using a linear-quadratic penalty algorithm*, ORSA J. Comput., 4, pp. 235–248.
- G. L. SCHULTZ AND R. R. MEYER (1991), *An interior point method for block angular optimization*, SIAM J. Optim., 1, pp. 583–602.
- J. TOMLIN (1966), *Minimum cost multicommodity network flows*, Oper. Res., 14, pp. 45–51.
- S. A. ZENIOS, M. C. PINAR, AND R. S. DEMBO (1995), *A smooth penalty function algorithm for network structured problems*, European J. Oper. Res., 83, pp. 220–236.

AN INTERIOR RANDOM VECTOR ALGORITHM FOR MULTISTAGE STOCHASTIC LINEAR PROGRAMS*

EITHAN SCHWEITZER†

Abstract. In this paper, an interior point algorithm for linear programs is adapted for solving multistage stochastic linear programs. The algorithm is based on Monteiro and Adler's path-following algorithm for deterministic linear programs. In practice, the complexity of the algorithm is linear with respect to the size of the sample space. The algorithm starts from a feasible solution of the problem and proceeds along a path of random vectors. The cubic polynomial complexity of the algorithm for deterministic linear programs is derived from the calculations of the Newton steps. In the algorithm developed in this paper, the probabilistic structure of the problem is taken into consideration while calculating a Newton step and the size of the sample space appears linearly in the complexity. The development of an algorithm that requires a relatively small number of arithmetic operations, in terms of the sample space size, allows the use of the algorithm for multistage stochastic linear programs with a very large number of scenarios.

Key words. stochastic programming, multistage linear programs, interior point methods

AMS subject classifications. 90C15, 90C05

PII. S105262349528456X

1. Introduction. Large-scale mathematical programming problems are often used to optimize long-period planning models. In most applications concerning planning, some parameters are not known with certainty. Long-period planning models must consider many possible outcomes of the future, called future *scenarios*. Describing the possible outcomes, or the future scenarios, becomes more difficult as the planning horizon grows, as does solving these problems. Multistage stochastic linear programs, introduced by Dantzig [7], [8] and Beale [1], are long-period mathematical programming problems in which some parameters (usually those of future stages) are not known with certainty when decisions with regard to the problem variables must be made. Multistage stochastic linear programs have a staircase structure, where all parameters with indices greater than 1 are random (see section 2).

If the number of future scenarios is finite, an exact deterministic equivalent of the stochastic program can be formulated and solved by conventional methods. In a practical multiperiod planning problem, the size of the resulting linear program, representing the deterministic equivalent, is often very large. The use of known interior point algorithms to solve the deterministic equivalent program requires a number of arithmetic operations with complexity that is at least cubic with respect to the number of future scenarios. In the multistage case, where the number of scenarios, say K , is large, executing $O(K^3)$ arithmetic operations is unrealistic. Birge and Holmes [5], Czyzyk, Fourer, and Mehrotra [6], and Lustig, Mulvey, and Carpenter [15] dealt with interior point algorithms for two-stage stochastic programming problems and achieved reduced complexity. Algorithms for two-stage stochastic linear programs can be used recursively for solving multistage stochastic linear programs. However, recursive use of two-stage algorithms for the multistage problem leads to algorithms with exponential

*Received by the editors April 12, 1995; accepted for publication (in revised form) September 11, 1997; published electronically August 26, 1998.

<http://www.siam.org/journals/siopt/8-4/28456.html>

†Faculty of Industrial Engineering and Management, Technion - Israel Institute of Technology, Haifa 32000, Israel (eithans@netvision.net.il).

complexity. Berger et al. [3] show reduced complexity for multistage stochastic linear programs that are augmented for coping with nonanticipativity.

The algorithm that will be introduced in this paper is based on Monteiro and Adler's path-following algorithm for deterministic linear programs; see [16] and [17]. In practice, the complexity of the algorithm is linear with respect to the size of the sample space. The algorithm starts from a feasible solution of the problem and proceeds along a path of random vectors. The cubic polynomial complexity is derived from the calculations of the Newton steps. In the algorithm developed in this paper, the probabilistic structure of the problem is taken into consideration while calculating a Newton step and the number of future scenarios appears linearly in the complexity bounds. The development of an algorithm that requires a relatively small number of arithmetic operations, in terms of the sample space size, allows the use of the algorithm for multistage stochastic linear programs with a very large number of scenarios. The algorithm can also be adapted to cases with infinitely many scenarios (by using Monte Carlo sampling, for example).

The multistage stochastic linear program is formulated in section 2, using terms of stochastic processes. The version of Monteiro and Adler's algorithm for multistage stochastic linear programs is described in sections 3 and 4. The convergence and the complexity of the algorithm is discussed in section 5. A practical improvement of the described algorithm is developed in section 6. Numerical results are given in section 7. It should be mentioned here that the algorithm is quite basic and a lot of improvements can be done by further research.

2. Problem formulation. In order to formulate a multistage stochastic linear problem, the probability space of the uncertain parameters is introduced first. Let $(\Omega, \mathcal{F}, \mathbf{P})$ be a probability space. Let $\{\mathcal{F}_t, t = 1, \dots, T\}$ be a filtration on Ω such that $\mathcal{F}_1 = \{\emptyset, \Omega\}$, where $t = 1, \dots, T$ is the stage number. For each $t = 1, \dots, T$, let A_t be an $m_t \times n_t$ random matrix, where m_t and n_t are constants. Suppose that A_t is of rank m_t $\mathbf{P}$ -a.s. Let B_t be an $m_t \times n_{t-1}$ random matrix. The random matrices $\{(A_t, B_{t-1}), t = 1, \dots, T\}$ are adapted to $\{\mathcal{F}_t, t = 1, \dots, T\}$ and are the coefficient matrices of the multistage stochastic linear problem at stage t . Let $\{b_t, t = 1, \dots, T\}$ be a sequence of m_t -dimensional random vectors adapted to $\{\mathcal{F}_t, t = 1, \dots, T\}$. Each b_t is the right-hand-side vector of the multistage stochastic linear problem at stage t . Let $\{c_t, t = 1, \dots, T\}$ be a sequence of n_t -dimensional random vectors adapted to $\{\mathcal{F}_t, t = 1, \dots, T\}$. Each c_t is the coefficients vector of the objective function at stage t . Since

$$(A_1, b_1, c_1) \in \mathcal{F}_1 = \{\emptyset, \Omega\},$$

they are deterministic and define the certain parameters of the present. The random variables $\{(A_t, B_{t-1}, c_t, b_t), t = 2, \dots, T\}$ define the uncertain parameters of the future. The probability space $(\Omega, \mathcal{F}, \mathbf{P})$ will be referred to as the probability space associated with the multistage stochastic linear problem.

In a general formulation of multistage stochastic linear problems, the objective is to find a deterministic *decision* (or strategy) $x_1 \in \mathcal{F}_1$ and a stochastic "*plan*" $x_t \in \mathcal{F}_t$ for each period $t = 2, \dots, T$. The stochastic process $\{x_t, t = 1, \dots, T\}$ should optimize the expected value of a linear function $\sum_{t=1}^T c'_t x_t$, where c'_t is the transpose of c_t . The vectors $\{x_t, t = 1, \dots, T\}$ are real valued n_t -dimensional nonnegative random vectors. The stochastic process $\{x_t, t = 1, \dots, T\}$ must satisfy, with probability one, the constraints

$$(2.1) \quad A_1 x_1 = b_1,$$

$$(2.2) \quad -B_{t-1}x_{t-1} + A_t x_t = b_t, \quad t = 2, \dots, T.$$

The general structure of the multistage stochastic linear program will then be

$$(2.3) \quad \begin{aligned} (\text{MSSLP}) \quad \min \quad & z = c'_1 x_1 + \mathbf{E}(c'_2 x_2 + \dots + c'_T x_T) \\ \text{s.t.} \quad & A_1 x_1 = b_1, \\ & -B_1 x_1 + A_2 x_2 = b_2, \\ & \ddots \quad \ddots \quad \vdots \\ & -B_{T-1} x_{T-1} + A_T x_T = b_T, \\ & x_1, x_2, \dots, x_T \geq 0, \\ & x_t \in \mathcal{F}_t, \quad t = 1, \dots, T. \end{aligned}$$

The random vectors $\{x_t, t = 1, \dots, T\}$ are adapted to the filtration generated by $\{(A_t, B_{t-1}, b_t, c_t), t = 1, \dots, T\}$. The requirement of x_t to be adapted is also known as nonanticipativity. In the formulation of (MSSLP) it means that for each realization of $\{(A_s, B_{s-1}, b_s, c_s), s = 1, \dots, T\}$, the value of x_t is a (measurable) function of solely $\{(A_s, B_{s-1}, b_s, c_s), s = 1, \dots, t\}$ but not $\{(A_s, B_{s-1}, b_s, c_s), s = t+1, \dots, T\}$.

The solution of (MSSLP) can be divided into two parts. The first part is the “here-and-now” decision x_1 and the second part is the “wait-and-see” solution $x_t, t = 2, \dots, T$, which is a plan for the future. If this solution exists, it defines a unique optimal value of the objective function.

To simplify notations, define, for each $\omega \in \Omega$,

$$(2.4) \quad A(\omega) = \begin{pmatrix} A_1 & & & \\ -B_1(\omega) & A_2(\omega) & & \\ & \ddots & \ddots & \\ & & -B_{T-1}(\omega) & A_T(\omega) \end{pmatrix}, \quad c(\omega) = \begin{pmatrix} c_1 \\ c_2(\omega) \\ \vdots \\ c_T(\omega) \end{pmatrix},$$

$$(2.5) \quad x(\omega) = \begin{pmatrix} x_1 \\ x_2(\omega) \\ \vdots \\ x_T(\omega) \end{pmatrix}, \quad b(\omega) = \begin{pmatrix} b_1 \\ b_2(\omega) \\ \vdots \\ b_T(\omega) \end{pmatrix},$$

and define $n = \sum_{t=1}^T n_t$ and $m = \sum_{t=1}^T m_t$.

In order to use the algorithm that will be described in this paper, assume that Ω is a finite set. If Ω is not finite, approximations can be made to estimate the solution of (MSSLP). Let $\mathcal{L}^n$ be the space of $\mathcal{F}$ -measurable functions mapping Ω into $\mathbf{R}^n$. Let

$$(2.6) \quad \mathcal{L}^n(\{\mathcal{F}_t\}) = \{x \in \mathcal{L}^n \mid x_t \in \mathcal{F}_t, t = 1, \dots, T\};$$

then $b \in \mathcal{L}^m(\{\mathcal{F}_t\})$ and $A \in \mathcal{L}^{m \times n}(\{\mathcal{F}_t\})$. With the above definitions, (MSSLP) can be written as

$$(2.7) \quad \begin{aligned} (\text{SP}) \quad \min \quad & \mathbf{E}(c'x) \\ \text{s.t.} \quad & Ax = b, \\ & x \geq 0, \\ & x \in \mathcal{L}^n(\{\mathcal{F}_t\}). \end{aligned}$$

With the assumption that Ω is finite, let $p_t(\omega)$, for each $\omega \in \Omega$, be the probability of a realization of the parameters generated by ω until stage t or the probability of

the subpath of ω until stage t . Note that each ω defines a path of possible future outcomes and can be written as

$$(2.8) \quad \omega = (\omega_1, \omega_2, \dots, \omega_T),$$

where ω_1 is a constant. Since $\mathcal{F}_t = \sigma\{\omega_1, \dots, \omega_t\}$, $p_t(\omega)$ can be written as

$$(2.9) \quad p_t(\omega) = \frac{\mathbf{P}(\omega)}{\mathbf{P}(\omega \mid \mathcal{F}_t)}.$$

Assume that $B_t \in \mathcal{F}_t$, $t = 1, \dots, T-1$. This assumption is essential to reduce the complexity of the interior random vector algorithm, as will be shown on page 963. In that case, the dual stochastic linear problem of (MSSLP), by taking the dual variable $\pi_t(\omega) = y_t(\omega)p_t(\omega)$ at stage t , is

$$(2.10) \quad \begin{array}{llll} \max & b'_1 y_1 & + & \mathbf{E}(b'_2 y_2) & + & \dots & + & b'_T y_T \\ \text{s.t.} & A'_1 y_1 - B'_1 \mathbf{E}(y_2 \mid \mathcal{F}_1) & & & & & & \leq c_1, \\ & & A'_2 y_2 - B'_2 \mathbf{E}(y_3 \mid \mathcal{F}_2) & & & & & \leq c_2, \\ & & & \ddots & & \ddots & & \vdots \\ & & & & & & A'_T y_T & \leq c_T, \end{array}$$

or, with the simplifying definitions,

$$(2.11) \quad \begin{array}{ll} \text{(SD)} & \max \quad \mathbf{E}(b'y) \\ & \text{s.t.} \quad A^\times y + z = c, \\ & \quad \quad z \geq 0, \\ & \quad \quad y \in \mathcal{L}^m(\{\mathcal{F}_t\}), \quad z \in \mathcal{L}^n(\{\mathcal{F}_t\}), \end{array}$$

where

$$(2.12) \quad A^\times(\omega) = \begin{pmatrix} A'_1 & -B'_1 \mathcal{E}_1 & & & \\ & A'_2(\omega)' & -B'_2(\omega)' \mathcal{E}_2 & & \\ & & \ddots & & \\ & & & & A'_T(\omega)' \end{pmatrix}$$

and $\mathcal{E}_t : \mathcal{L}^{m_{t+1}}(\mathcal{F}_{t+1}) \rightarrow \mathcal{L}^{m_{t+1}}(\mathcal{F}_t)$ is the conditional expectation defined as $\mathcal{E}_t(y_{t+1}) = \mathbf{E}(y_{t+1} \mid \mathcal{F}_t)$.

With this notation of $A^\times$, for any feasible solution x of (MSSLP),

$$(2.13) \quad \mathbf{E}(x' A^\times y) = \mathbf{E}(b'y) = \mathbf{E}(y' A x).$$

Assume, as before, that Ω is finite and let

$$(2.14) \quad 1 = K_1 \leq K_2 \leq \dots \leq K_T = K = |\Omega|,$$

where K_t is the total number of realizations of all parameters until stage t . The size of the deterministic equivalent linear program of (MSSLP) is $M \times N$, where

$$(2.15) \quad M = \sum_{t=1}^T m_t K_t \leq mK,$$

$$(2.16) \quad N = \sum_{t=1}^T n_t K_t \leq nK.$$

3. The algorithm. The use of the logarithmic barrier function method (see Fiacco and McCormick [10]) for the stochastic linear program (SP), as was defined in (2.7), will provide a family of problems

$$(3.1) \quad \begin{aligned} (\text{SP}_\mu) \quad & \min \quad \mathbf{E} \left(c'x - \mu \sum_{j=1}^n q_j \ln x_j \right) \\ & \text{s.t.} \quad Ax = b, \\ & \quad x > 0, \end{aligned}$$

where $\mu > 0$ is the barrier penalty parameter and

$$(3.2) \quad q_j(\omega) = p_t^{-1}(\omega) \quad \text{for all } \sum_{s=1}^{t-1} n_s < j \leq \sum_{s=1}^t n_s.$$

With this definition of q , the function $-\mathbf{E}(\sum_{j=1}^n q_j \ln x_j)$ is an N -self concordant barrier and the polynomial convergence results that were mentioned are valid (see Nesterov and Nemirovskii [20]).

In order to apply the logarithmic barrier function method, it should be assumed that the interior of the feasible set for (SP) and of the feasible set for (SD),

$$(3.3) \quad \mathcal{S}^0 = \{x \in \mathcal{L}^n(\{\mathcal{F}_t\}) \mid Ax = b, x > 0 \ \forall \omega \in \Omega\},$$

$$(3.4) \quad \mathcal{T}^0 = \{(y, z) \in \mathcal{L}^m(\{\mathcal{F}_t\}) \times \mathcal{L}^n(\{\mathcal{F}_t\}) \mid A^\times y + z = c, z > 0 \ \forall \omega \in \Omega\},$$

respectively, are not empty. (Note that since Ω is assumed to be finite, $\forall \omega \in \Omega$ is equivalent to $\mathbf{P}$ -a.s.) Since the objective of (SP_μ) is to minimize a strictly convex function over an open set, $x = x(\mu) \in \mathcal{S}^0$ is the optimal solution of (SP_μ) if and only if there exists $(u, y) = (u(\mu), y(\mu))$, $u \geq 0 \ \forall \omega \in \Omega$, such that

$$(3.5) \quad \begin{aligned} & \mathbf{E}(u'x) = 0, \\ & c - \mu QX^{-1}e - A^\times y = 0 \quad \forall \omega \in \Omega, \\ & Ax = b, \ x > 0, \\ & (x, u, y) \in \mathcal{L}^n(\{\mathcal{F}_t\}) \times \mathcal{L}^n(\{\mathcal{F}_t\}) \times \mathcal{L}^m(\{\mathcal{F}_t\}), \end{aligned}$$

where $X = \text{diag}(x_1, \dots, x_n)$, $Q = \text{diag}(q_1, \dots, q_n)$, and e is a vector of ones in the appropriate dimension. Note that in the solution of (3.5), $u \geq 0$, $x > 0$, and $\mathbf{E}(u'x) = 0$ imply that $u = 0 \ \forall \omega \in \Omega$.

Let $z = c - A^\times y$ and define $Z = \text{diag}(z_1, \dots, z_n)$. Then (3.5) can be written as

$$(3.6) \quad \begin{aligned} & ZXe = \mu Qe, \\ & Ax = b, \ x > 0, \quad \forall \omega \in \Omega, \\ & A^\times y + z = c, \\ & (x, y, z) \in \mathcal{L}^n(\{\mathcal{F}_t\}) \times \mathcal{L}^m(\{\mathcal{F}_t\}) \times \mathcal{L}^n(\{\mathcal{F}_t\}). \end{aligned}$$

For a given $\mu > 0$, $ZXe = \mu Qe$, $x > 0$ imply that $z > 0 \ \forall \omega \in \Omega$. That, with the equation $A^\times y + z = c$, implies that $(y, z) \in \mathcal{T}^0$, which is assumed to be nonempty. Since (SP_μ) has a unique solution x , z is also unique. For that unique z , $A^\times y = c - z$. Since $\text{rank}(A) = m$ for all $\omega \in \Omega$, the rows of A are linearly independent and the homogeneous system $A^\times y = 0$ has no solution other than zero. Therefore, for each $\mu > 0$, there exists a unique random vector $(x, y, z) = (x(\mu), y(\mu), z(\mu))$ that satisfies (3.6).

For each $\mu > 0$, the duality gap at that point is

$$\begin{aligned}
 (3.7) \quad g(x(\mu), y(\mu), z(\mu)) &= \mathbf{E}(c'x(\mu) - b'y(\mu)) \\
 &= \mathbf{E}(c'x(\mu) - x'A^\times y(\mu)) = \mathbf{E}((c - A^\times y(\mu))'x(\mu)) \\
 &= \mathbf{E}(z(\mu)'x(\mu)) = \mathbf{E}(e'Z(\mu)X(\mu)e) \\
 &= \mu \mathbf{E}(e'Qe) = \mu \sum_{t=1}^T n_t K_t \\
 &= N\mu \quad \forall \mu > 0.
 \end{aligned}$$

Hence,

$$(3.8) \quad g(x(\mu), y(\mu), z(\mu)) \rightarrow 0 \quad \text{as } \mu \rightarrow 0.$$

Following Monteiro and Adler's notation, the definition of the central path associated with (SP) is

$$\Gamma = \{(x(\mu), y(\mu), z(\mu)) \mid \mu > 0, (x(\mu), y(\mu), z(\mu)) \text{ solves (3.6) for } \mu\}.$$

Let

$$H(x, y, z|\mu) = \begin{pmatrix} ZXe - \mu Qe \\ Ax - b \\ A^\times y + z - c \end{pmatrix}.$$

For every given $\mu > 0$, it is desired to solve $H(x, y, z|\mu) = 0$. The Newton step $(\Delta x, \Delta y, \Delta z)$ at the point (x, y, z) is determined by the system of linear equations

$$(3.9) \quad J_H(x, y, z) \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = H(x, y, z|\mu),$$

where

$$(3.10) \quad J_H(x, y, z) = \begin{pmatrix} Z & 0 & X \\ A & 0 & 0 \\ 0 & A^\times & I \end{pmatrix}$$

is the Jacobian of H at (x, y, z) . After calculating the Newton step, the algorithm proceeds from the point (x, y, z) to a new point $(x + \Delta x, y + \Delta y, z + \Delta z)$. The new point is also a random vector, and in order to keep measurability it is required that

$$\Delta x \in \mathcal{L}^n(\{\mathcal{F}_t\}), \quad \Delta y \in \mathcal{L}^m(\{\mathcal{F}_t\}), \quad \Delta z \in \mathcal{L}^n(\{\mathcal{F}_t\}).$$

The linear system (3.9) has a unique solution and has to be solved for all $\omega \in \Omega$. Expectations and random variables are mixed up in this system. Moreover, some of the variables in that system are deterministic, some are $\mathcal{F}_2$ -measurable, some are $\mathcal{F}_3$ -measurable, etc. The linear system can be written and solved as a huge deterministic linear system. However, when K is large, an attempt to solve such a system will require $O(n^3 K^3)$ arithmetic operations, which may be an enormous number. Fortunately, by variable substitutions, it is possible to solve this linear system efficiently. In the algorithm that follows, K appears linearly in the complexity. The main idea in the following algorithm is to solve (3.9) conditioned on $\mathcal{F}_t$ for $t = 1, \dots, T$.

For each vector in the interior of the feasible set $\mathcal{S}^0 \times \mathcal{T}^0$, the linear system (3.9) can be written as

$$(3.11) \quad \begin{pmatrix} Z & 0 & X \\ A & 0 & 0 \\ 0 & A^\times & I \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = \begin{pmatrix} ZXe - \mu Qe \\ 0 \\ 0 \end{pmatrix}.$$

Since x and z are in the interior of the feasible set, X and Z are diagonal matrices with strictly positive elements in the diagonal. Hence, they are invertible and, with the assumption that the coefficient matrices in (SP) are all full ranked, the linear system (3.11) has a unique solution.

The linear system (3.11) can be written as a set of three linear systems,

$$(3.12) \quad Z\Delta x + X\Delta z = ZXe - \mu Qe,$$

$$(3.13) \quad A\Delta x = 0,$$

$$(3.14) \quad A^\times \Delta y + \Delta z = 0.$$

From (3.12),

$$(3.15) \quad \Delta x = -Z^{-1}X\Delta z + Xe - \mu QZ^{-1}e.$$

From (3.14),

$$(3.16) \quad \Delta z = -A^\times \Delta y.$$

By substituting (3.16) into (3.15) it is obtained that

$$(3.17) \quad \Delta x = Z^{-1}XA^\times \Delta y + Xe - \mu QZ^{-1}e.$$

Therefore, if Δy is known, Δx and Δz can be easily calculated.

If (3.17) is substituted in (3.13), the following linear system is obtained:

$$(3.18) \quad AZ^{-1}XA^\times \Delta y + AXe - \mu AQZ^{-1}e = 0.$$

Note that $AXe = b$ and obtain

$$(3.19) \quad AZ^{-1}XA^\times \Delta y + b - \mu AQZ^{-1}e = 0.$$

The solution of (3.19) is the required Δy .

The linear system (3.19) is a multistage stochastic linear equations system in terms of Δy . In order to see that, define

$$(3.20) \quad U_t = A_t Z_t^{-1} X_t A'_t, \quad t = 1, \dots, T,$$

$$(3.21) \quad V_t = A_t Z_t^{-1} X_t B'_t, \quad t = 1, \dots, T-1, \quad V_0 = 0, \quad V_T = 0,$$

$$(3.22) \quad W_t = B_t Z_t^{-1} X_t B'_t, \quad t = 1, \dots, T-1, \quad W_0 = 0, \quad W_T = 0,$$

$$(3.23) \quad r_t = b_t - \mu A_t Q_t Z_t^{-1} e, \quad t = 1, \dots, T,$$

$$(3.24) \quad s_t = \mu B_t Q_t Z_t^{-1} e, \quad t = 1, \dots, T-1, \quad s_0 = 0, \quad s_T = 0.$$

Then, expand (3.19) using these definitions to obtain the multistage stochastic linear equations system

$$(3.25) \quad \begin{array}{ll} U_1 \Delta y_1 - V_1 \mathcal{E}_1(\Delta y_2) & + r_1 = 0, \\ -V'_1 \Delta y_1 + W_1 \mathcal{E}_1(\Delta y_2) + U_2 \Delta y_2 - V_2 \mathcal{E}_2(\Delta y_3) & + r_2 + s_1 = 0, \\ \vdots & \vdots \\ -V'_{t-1} \Delta y_{t-1} + W_{t-1} \mathcal{E}_{t-1}(\Delta y_t) + U_t \Delta y_t - V_t \mathcal{E}_t(\Delta y_{t+1}) & + r_t + s_{t-1} = 0, \\ \vdots & \vdots \\ -V'_{T-1} \Delta y_{T-1} + W_{T-1} \mathcal{E}_{T-1}(\Delta y_T) + U_T \Delta y_T & + r_T + s_{T-1} = 0. \end{array}$$

Define

$$(3.26) \quad D_T = U_T.$$

From the constraints of (3.25) related to time T ,

$$(3.27) \quad \Delta y_T = D_T^{-1} [V'_{T-1} \Delta y_{T-1} - W_{T-1} \mathcal{E}_{T-1}(\Delta y_T) - r_T - s_{T-1}].$$

Let

$$(3.28) \quad G_{T-1} = I + \mathcal{E}_{T-1}(D_T^{-1})W_{T-1},$$

$$(3.29) \quad h_{T-1} = \mathcal{E}_{T-1}(D_T^{-1}r_T) + \mathcal{E}_{T-1}(D_T^{-1})s_{T-1}.$$

Take expectations, conditioned on $\mathcal{F}_{T-1}$, on both sides of (3.27); then substitute (3.28) and (3.29) into the resulting equation to obtain

$$(3.30) \quad \mathcal{E}_{T-1}(\Delta y_T) = G_{T-1}^{-1} [\mathcal{E}_{T-1}(D_T^{-1})V'_{T-1} \Delta y_{T-1} - h_{T-1}].$$

Note that the restriction $B_t \in \mathcal{F}_t$, $t = 1, \dots, T-1$, implies that $\mathcal{E}_{T-1}(W_{T-1} \Delta y_T) = W_{T-1} \mathcal{E}_{T-1}(\Delta y_T)$. Without that restriction, an explicit expression for $\mathcal{E}_{T-1}(\Delta y_T)$ could not be obtained.

Substitute (3.30) into the constraints of (3.25) related to time $T-1$ to obtain

$$(3.31) \quad -V'_{T-2} \Delta y_{T-2} + W_{T-2} \mathcal{E}_{T-2}(\Delta y_{T-1}) + U_{T-1} \Delta y_{T-1} \\ - V_{T-1} G_{T-1}^{-1} \mathcal{E}_{T-1}(D_T^{-1})V'_{T-1} \Delta y_{T-1} + V_{T-1} G_{T-1}^{-1} h_{T-1} + r_{T-1} + s_{T-2} = 0.$$

Define

$$(3.32) \quad D_{T-1} = U_{T-1} - V_{T-1} G_{T-1}^{-1} \mathcal{E}_{T-1}(D_T^{-1})V'_{T-1};$$

hence,

$$(3.33) \quad \Delta y_{T-1} = D_{T-1}^{-1} [V'_{T-2} \Delta y_{T-2} - W_{T-2} \mathcal{E}_{T-2}(\Delta y_{T-1}) \\ - V_{T-1} G_{T-1}^{-1} h_{T-1} - r_{T-1} - s_{T-2}].$$

Let

$$(3.34) \quad G_{T-2} = I + \mathcal{E}_{T-2}(D_{T-1}^{-1})W_{T-2},$$

$$h_{T-2} = \mathcal{E}_{T-2}(D_{T-1}^{-1}V_{T-1}G_{T-1}^{-1}h_{T-1}) + \mathcal{E}_{T-2}(D_{T-1}^{-1}r_{T-1})$$

$$(3.35) \quad + \mathcal{E}_{T-2}(D_{T-1}^{-1})s_{T-2}.$$

Take expectations, conditioned on $\mathcal{F}_{T-2}$, on both sides of (3.33), then substitute (3.34) and (3.35) into the resulting equation to obtain

$$(3.36) \quad \mathcal{E}_{T-2}(\Delta y_{T-1}) = G_{T-2}^{-1} [\mathcal{E}_{T-2}(D_{T-1}^{-1})V'_{T-2} \Delta y_{T-2} - h_{T-2}],$$

and so on. The algorithm to solve (3.11) is given as follows.

ALGORITHM 1.

Set $D_T = U_T$, $G_T = 0$, $h_T = 0$, $\Delta y_0 = 0$, $\mathcal{E}_0(\Delta y_1) = 0$.
For $t = T-1$ down to 1, set

$$G_t = I + \mathcal{E}_t(D_{t+1}^{-1})W_t,$$

$$h_t = \mathcal{E}_t(D_{t+1}^{-1}V_{t+1}G_{t+1}^{-1}h_{t+1}) + \mathcal{E}_t(D_{t+1}^{-1}r_{t+1}) + \mathcal{E}_t(D_{t+1}^{-1})s_t,$$

$$D_t = U_t - V_t G_t^{-1} \mathcal{E}_t(D_{t+1}^{-1})V'_t.$$

For $t = 1, \dots, T$, set

$$\begin{aligned}\Delta y_t &= D_t^{-1}[V'_{t-1}\Delta y_{t-1} - W_{t-1}\mathcal{E}_{t-1}(\Delta y_t) - V_t G_t^{-1}h_t - r_t - s_{t-1}], \\ \mathcal{E}_t(\Delta y_{t+1}) &= G_t^{-1}[\mathcal{E}_t(D_{t+1}^{-1})V'_t\Delta y_t - h_t] \quad (\text{for } t < T), \\ \Delta z_t &= -A'_t\Delta y_t + B'_t\mathcal{E}_t(\Delta y_{t+1}), \\ \Delta x_t &= -Z_t^{-1}X_t\Delta z_t + Xe - \mu Q_t Z_t^{-1}e.\end{aligned}$$

Note that the matrices G_t and D_t and the vectors h_t are all given by the data of the problem, and that all variables with index t are $\mathcal{F}_t$ -measurable. Using the Sherman–Morrison–Woodbury formula, it is easy to see that the matrices $\{G_t, t = 1, \dots, T-1\}$ are all invertible and the matrices $\{D_t, t = 1, \dots, T\}$ are all positive definite and hence invertible.

The main idea of the path-following algorithm is to produce a sequence of points that are close to the path Γ and that converges to the optimal solutions of (SP) and (SD). Monteiro and Adler had defined closeness by the proximity condition

$$\|ZXe - \mu e\| \leq \theta\mu.$$

In the stochastic programming case as defined above, $\|x\| = \mathbf{E}^{\frac{1}{2}}(x'x)$. However, using the proximity condition defined by that $\mathcal{L}_2$ norm may cause the algorithm to produce points that are too far from the central path. A better definition for the proximity condition for stochastic programs can be

$$(3.37) \quad \|XZe - \mu Qe\|_P \leq \theta\mu,$$

where $\|x\|_P = \mathbf{E}^{\frac{1}{2}}(x'Px)$, $P = Q^{-1}$.

The parameter θ is one of two constant parameters θ and δ , defined by Monteiro and Adler, that satisfy

$$(3.38) \quad \begin{aligned}0 &\leq \theta < \frac{1}{2}, \quad 0 < \delta < \frac{1}{\sqrt{2}}, \\ \frac{\theta^2 + \delta^2}{2(1-\theta)} &\leq \theta(1 - \frac{\delta}{\sqrt{N}}).\end{aligned}$$

A good choice of δ and θ is $\delta = 0.42$, $\theta = 0.183$. At the beginning of the algorithm it is assumed that an initial feasible point that satisfies (3.37) is provided. The way to find this first interior random vector will be described in the next section. The algorithm is defined as follows.

ALGORITHM 2.

Given: θ and δ that satisfy (3.38).

A feasible random vector (x^0, y^0, z^0) and a positive number μ_0 that satisfy the proximity condition (3.37).

$\epsilon > 0$, the tolerance for the duality gap.

Set $k := 0$.

While $\mathbf{E}(x^{k'}z^k) > \epsilon$,

$$\mu_{k+1} = \mu_k(1 - \frac{\delta}{\sqrt{N}}).$$

Use Algorithm 1 to calculate $(\Delta x^k, \Delta y^k, \Delta z^k)$ by

$$J_H(x^k, y^k, z^k) \begin{pmatrix} \Delta x^k \\ \Delta y^k \\ \Delta z^k \end{pmatrix} = H(x^k, y^k, z^k | \mu_{k+1}).$$

Set $x^{k+1} = x^k - \Delta x^k$,

$$\begin{aligned} y^{k+1} &= y^k - \Delta y^k, \\ z^{k+1} &= z^k - \Delta z^k. \end{aligned}$$

$$k := k + 1.$$

End while.

Monteiro and Adler had shown that, in the deterministic case, Algorithm 2 terminates in at most $O(\sqrt{N} \log N)$ iterations (when ϵ and μ_0 are considered as constants). In most cases, μ can be updated by a constant factor and the total number of iterations depends only on $\log N$ which is not so large. In section 6, a version of this algorithm will be introduced. In that version, the algorithm will be terminated in at most $O(N \log n)$ iterations, which is worse than the above. However, this is only in the worst case analysis. In practice, the number of iterations will depend only on $\log n$, which is smaller than $\log N$.

4. Initializing the algorithm. Algorithm 2 requires an initial feasible point that satisfies the proximity condition (3.37). Monteiro and Adler have shown how to choose the initial point in the deterministic case. Their method is helpful also in the multistage stochastic programming case and makes it clear why the coefficient matrices $A_2, \dots, A_T$ cannot be assumed deterministic in the construction of the algorithm.

Consider the multistage stochastic linear program

$$\begin{aligned} (\tilde{\text{SP}}) \quad & \min \quad \mathbf{E}(\tilde{c}'\tilde{x}) \\ & \text{s.t.} \quad \tilde{A}\tilde{x} = \tilde{b}, \\ & \quad \tilde{x} \geq 0, \end{aligned}$$

where its elements are defined as in (2.4)–(2.5). With these notations, define

$$L = \left\lceil \log \left(1 + \max_{\omega \in \Omega} \left\{ \max_j |p_i(\omega) \tilde{c}_j(\omega)|, \max_i |p_i(\omega) \tilde{b}_i(\omega)|, \max_{i,j} |p_i(\omega) \tilde{A}_{i,j}(\omega)|, \tilde{m}, \tilde{n} \right\} \right) \right\rceil.$$

(This is *not* the input length. The defined L is, generally, the largest value in the problem and is used as a penalty parameter.)

$$\begin{aligned} n_t &= \tilde{n}_t + 2, \quad m_t = \tilde{m}_t + 1, \quad t = 1, \dots, T, \\ \alpha &= 2^{4L}, \quad \lambda = 2^{2L}, \quad \lambda_t = \lambda q_t, \\ K_{b_t} &= \alpha \lambda_t (\tilde{n}_t + 1) - \lambda_t \tilde{c}_t' e, \quad K_{c_t} = \alpha \lambda_t, \quad t = 1, \dots, T. \end{aligned}$$

Then, for $t = 1, \dots, T$, let

$$(4.1) \quad A_t = \begin{pmatrix} \tilde{A}_t & 0 & \tilde{b}_t - \lambda_t \tilde{A}_t e \\ (\alpha e - \tilde{c}_t)' & \alpha & 0 \end{pmatrix},$$

$$(4.2) \quad B_t = \begin{pmatrix} \tilde{B}_t & 0 & -\lambda_t \tilde{B}_t e \\ 0 & 0 & 0 \end{pmatrix},$$

$$(4.3) \quad c_t = \begin{pmatrix} \tilde{c}_t \\ 0 \\ K_{c_t} \end{pmatrix}, \quad b_t = \begin{pmatrix} \tilde{b}_t \\ K_{b_t} \end{pmatrix}.$$

Note that $(A_t, B_{t-1}, c_t, b_t) \in \mathcal{F}_t$ for $t = 1, \dots, T$. With those definitions, consider the multistage stochastic linear program (SP), where its elements are defined by (4.1)–(4.3) as in (2.4)–(2.5).

For $t = 1, \dots, T$, let

$$\begin{aligned} x_t^0 &= (\lambda_t, \lambda_t, \dots, \lambda_t, 1)' \in \mathbf{R}^{n_t} \quad \forall \omega \in \Omega, \\ y_t^0 &= (0, 0, \dots, 0, -1)' \in \mathbf{R}^{m_t} \quad \forall \omega \in \Omega, \\ z_t^0 &= (\alpha, \alpha, \dots, \alpha, \alpha \lambda_t)' \in \mathbf{R}^{n_t} \quad \forall \omega \in \Omega. \end{aligned}$$

Clearly, $x_t^0 > 0$ and $z_t^0 > 0 \forall \omega \in \Omega$ for all $t = 1, \dots, T$,

$$\begin{aligned} A_1 x_1^0 &= \begin{pmatrix} \lambda_1 \tilde{A}_1 e + \tilde{b}_1 - \lambda_1 \tilde{A}_1 e \\ \lambda_1 (\alpha e - \tilde{c}_1)' e + \alpha \lambda_1 \end{pmatrix} = \begin{pmatrix} \tilde{b}_1 \\ \alpha \lambda_1 (\tilde{n}_1 + 1) - \lambda_1 \tilde{c}_1' e \end{pmatrix} = b_1, \\ -B_{t-1} x_{t-1}^0 + A_t x_t^0 &= \begin{pmatrix} -\lambda_{t-1} \tilde{B}_{t-1} e + \lambda_{t-1} \tilde{B}_{t-1} e + \lambda_t \tilde{A}_t e + \tilde{b}_t - \lambda_t \tilde{A}_t e \\ \lambda_t (\alpha e - \tilde{c}_t)' e + \alpha \lambda_t \end{pmatrix} \\ &= \begin{pmatrix} \tilde{b}_t \\ \alpha \lambda_t (\tilde{n}_t + 1) - \lambda_t \tilde{c}_t' e \end{pmatrix} = b_t, \quad t = 2, \dots, T, \\ A'_t y_t^0 - B'_t \mathbf{E}(y_{t+1}^0 | \mathcal{F}_t) + z_t^0 &= \begin{pmatrix} -(\alpha e - \tilde{c}_t) & + & \alpha e \\ -\alpha & + & \alpha \\ 0 & + & \alpha \lambda_t \end{pmatrix} = \begin{pmatrix} \tilde{c}_t \\ 0 \\ K_{c_t} \end{pmatrix} = c_t, \\ &\quad t = 1, \dots, T-1, \\ A'_T y_T^0 + z_T^0 &= \begin{pmatrix} -(\alpha e - \tilde{c}_T) & + & \alpha e \\ -\alpha & + & \alpha \\ 0 & + & \alpha \lambda_T \end{pmatrix} = \begin{pmatrix} \tilde{c}_T \\ 0 \\ K_{c_T} \end{pmatrix} = c_T. \end{aligned}$$

Therefore, the feasible set of (SP) and of its dual are nonempty and (x^0, y^0, z^0) satisfies

$$\begin{aligned} Ax^0 &= b, \quad x^0 > 0, \\ A^\times y^0 + z^0 &= c, \quad z^0 > 0, \end{aligned}$$

with

$$X^0 Z^0 e = \alpha \lambda Q e.$$

Taking

$$\mu_0 = \frac{\mathbf{E}(x^0 z^0)}{N} = \alpha \lambda$$

will satisfy the proximity condition

$$\|X^0 Z^0 e - \mu_0 Q e\|_P^2 = 0 \leq \theta^2 \mu_0^2,$$

since (x^0, y^0, z^0) is a point on the central path in $\mathcal{L}^n(\{\mathcal{F}_t\}) \times \mathcal{L}^m(\{\mathcal{F}_t\}) \times \mathcal{L}^n(\{\mathcal{F}_t\})$.

5. Convergence and complexity. In this section, the convergence and the complexity of Algorithm 2 will be discussed. The convergence results are extensions of Monteiro and Adler's results for the deterministic case.

To prove convergence, substitute the above P -norm in the proofs of the convergence of Monteiro and Adler's algorithms to obtain the proofs of the following convergence theorem of Algorithm 2. The following theorem proves that if (x, y, z) is a feasible random vector that satisfies the proximity condition with respect to μ ,

then $(\hat{x}, \hat{y}, \hat{z})$ is also a feasible random vector that satisfies the proximity condition with respect to $\hat{\mu} < \mu$. It also proves that the duality gap at $(\hat{x}, \hat{y}, \hat{z})$ is $N\hat{\mu}$, which is smaller than the duality gap at the previous point.

THEOREM 5.1. *Let θ and δ be constants that satisfy (3.38). Let (x, y, z) be a random vector such that $x \in \mathcal{S}^0$, $(y, z) \in \mathcal{T}^0$ and that satisfies $\|XZe - \mu Qe\|_P \leq \theta\mu$. Let $\hat{\mu} = \mu(1 - \frac{\delta}{\sqrt{N}})$ and $\hat{x} = x - \Delta x$, $\hat{y} = y - \Delta y$, $\hat{z} = z - \Delta z$, where $(\Delta x, \Delta y, \Delta z)$ is the solution of (3.11) with the parameter $\hat{\mu}$. Then*

- (a) $\|\hat{X}\hat{Z}e - \hat{\mu}Qe\|_P \leq \theta\hat{\mu}$.
- (b) $\hat{x} \in \mathcal{S}^0$, $(\hat{y}, \hat{z}) \in \mathcal{T}^0$.
- (c) $g(\hat{x}, \hat{y}, \hat{z}) = \mathbf{E}(\hat{x}'\hat{z}) = N\hat{\mu}$.

By the construction of the initial point, (x^0, y^0, z^0) satisfies the condition of Theorem 5.1. Theorem 5.1 is then the basis for the induction to prove that the sequence of points generated by Algorithm 2 satisfies (a), (b), and (c) of that theorem, hence the following.

COROLLARY 5.2. *The sequence of points (x^k, y^k, z^k) generated by Algorithm 2 satisfies, for all $k = 1, 2, \dots$, the relations*

- (a) $\|X^k Z^k e - \mu_k Qe\|_P \leq \theta\mu_k$.
- (b) $x^k \in \mathcal{S}^0$, $(y^k, z^k) \in \mathcal{T}^0$.
- (c) $g(x^k, y^k, z^k) = \mathbf{E}(x^{k'} z^k) = N\mu_k$.

For the discussion of complexity, refer to Proposition 4.1 in [16].

PROPOSITION 5.3. *The total number of iterations performed by Algorithm 2 is no greater than*

$$\hat{k} = \left\lceil \log(N\epsilon^{-1}\mu_0) \frac{\sqrt{N}}{\delta} \right\rceil,$$

where $\epsilon > 0$ denotes the tolerance of the duality gap and $\mu_0 = \frac{\mathbf{E}(x^{0'} z^0)}{N}$.

Since $\mu_0 = 2^{6L}$,

$$\hat{k} \leq 1 + \log(N\epsilon^{-1}2^{6L}) \frac{\sqrt{N}}{\delta} = 1 + (\log N - \log \epsilon + 6L) \frac{\sqrt{N}}{\delta}.$$

Therefore, $\hat{k}$ is of $O(\sqrt{N} \log N)$ (for given ϵ and L).

In order to check the total complexity of Algorithm 2, it is required to count the number of arithmetic operations at every iteration. According to Algorithm 1, for each t and for each ω , the computational effort is dominated by the calculation of D_{t+1}^{-1} , G_t^{-1} , and the matrix multiplications, that require $O(n_{t+1}^3)$ arithmetic operations, since $m_t \leq n_t$ for all $t = 1, \dots, T$. Thus, for each t , the number of arithmetic operations is of

$$O(n_{t+1}^3 K_{t+1}).$$

Therefore, the total number of arithmetic operations in one iteration is of

$$O\left(\sum_{t=1}^T n_t^3 K_t\right) \leq O(n^3 K),$$

and the total number of arithmetic operations required by the algorithm is bounded by $O(Kn^3\sqrt{N} \log N)$.

6. A practical improvement. In this section, Algorithm 2 will be modified by taking $q_i(\omega) = 1$ for all i and for all ω . This modification will give a worst case complexity that is larger than the complexity of Algorithm 2. It will need $O(N \log n)$ iteration, while Algorithm 2 needs $O(\sqrt{N} \log N)$ iterations (for given ϵ and L). However, in many practical cases, it is possible in both algorithms to update μ by a constant factor and to get faster convergence. In these cases, the number of iterations that are needed by Algorithm 2 is of $O(\log N)$, while the modified algorithm will use $O(\log n)$ iterations.

Let $q_i(\omega) = 1$ for all $i = 1, \dots, n$, $\omega \in \Omega$. With this definition, system (3.6) will be

$$(6.1) \quad \begin{aligned} ZXe &= \mu e, \\ Ax &= b, \quad x > 0, \quad \forall \omega \in \Omega, \\ A^\times y + z &= c, \\ (x, y, z) &\in \mathcal{L}^n(\{\mathcal{F}_t\}) \times \mathcal{L}^m(\{\mathcal{F}_t\}) \times \mathcal{L}^n(\{\mathcal{F}_t\}). \end{aligned}$$

For each $\mu > 0$, the duality gap at a point $(x, y, z) = (x(\mu), y(\mu), z(\mu))$ that satisfies (6.1) is

$$(6.2) \quad \begin{aligned} g(x(\mu), y(\mu), z(\mu)) &= \mathbf{E}(c'x(\mu) - b'y(\mu)) \\ &= \mathbf{E}(c'x(\mu) - x'A^\times y(\mu)) = \mathbf{E}((c - A^\times y(\mu))'x(\mu)) \\ &= \mathbf{E}(z(\mu)'x(\mu)) = \mathbf{E}(e'Z(\mu)X(\mu)e) \\ &= n\mu \quad \forall \mu > 0. \end{aligned}$$

By taking (x^0, y^0, z^0) as in section 4,

$$\mu_0 = \frac{\mathbf{E}(x^{0'} z^0)}{n} = \alpha \lambda.$$

Algorithm 2 and the modified algorithm start with the same $\mu_0 = 2^{6L}$ but, according to (6.2), the modified algorithm will start with a smaller duality gap. If μ is updated by a constant factor, the modified algorithm will converge faster.

Define new vectors on $\mathbf{R}^N$ in the following way: let $u \in \mathbf{R}^N$ be the vector of all the elements $x_{t,i}(\omega)$; let $v \in \mathbf{R}^N$ be the vector of all the elements $p_t(\omega)z_{t,i}(\omega)$; let P be an $N \times N$ diagonal matrix with the elements $p_t(\omega)$ appearing n_t times diagonally.

The starting point, (x^0, y^0, z^0) , will not satisfy the proximity condition (3.37), but will satisfy a new property,

$$(6.3) \quad \|UVe - \mu Pe\| \leq \theta \mu,$$

where $\|\cdot\|$ is the Euclidean norm in this case. Note that

$$(6.4) \quad XZe = \mu e \quad \forall \omega \in \Omega \quad \text{if and only if} \quad UVe = \mu Pe.$$

Let

$$\bar{H}(x, y, z|\mu) = \begin{pmatrix} ZXe - \mu e \\ Ax - b \\ A^\times y + z - c \end{pmatrix}.$$

The modified algorithm will then be the following.

ALGORITHM 3.

Given: θ and δ that satisfy (6.5) (that will be defined later).

A feasible random vector (x^0, y^0, z^0) and a positive number μ_0 that satisfy the proximity condition (6.3).

$\epsilon > 0$, the tolerance for the duality gap.

Set $k := 0$.

While $\mathbf{E}(x^{k'} z^k) > \epsilon$,

$$\mu_{k+1} = \mu_k \left(1 - \frac{\delta}{\sqrt{n}}\right).$$

Use Algorithm 1 to calculate $(\Delta x^k, \Delta y^k, \Delta z^k)$ by

$$J_{\bar{H}}(x^k, y^k, z^k) \begin{pmatrix} \Delta x^k \\ \Delta y^k \\ \Delta z^k \end{pmatrix} = \bar{H}(x^k, y^k, z^k | \mu_{k+1}).$$

$$\begin{aligned} \text{Set } x^{k+1} &= x^k - \Delta x^k, \\ y^{k+1} &= y^k - \Delta y^k, \\ z^{k+1} &= z^k - \Delta z^k. \end{aligned}$$

$$k := k + 1.$$

End while.

In order to see that Algorithm 3 is correct and to check the number of iterations needed, substitute the P -norm of section 5 to the Euclidean norm. Note that the number of arithmetic operations that are needed by Algorithm 1, used in both Algorithms 2 and 3, is not changed.

Let (x, y, z) be a random vector produced by Algorithm 3 in a certain iteration with μ . Let $\hat{\mu} = \mu(1 - \frac{\delta}{\sqrt{n}})$ and let $\hat{x} = x - \Delta x, \hat{y} = y - \Delta y, \hat{z} = z - \Delta z$ be the next random vector produced by Algorithm 3. The following results refer to Algorithm 3.

A modification of Theorem 5.1 is as follows.

THEOREM 6.1. *Let θ and δ be constants that satisfy (6.5). Let (x, y, z) be a random vector such that $x \in \mathcal{S}^0$, $(y, z) \in \mathcal{T}^0$ and that satisfies $\|UVe - \mu Pe\| \leq \theta\mu$. Let $\hat{\mu} = \mu(1 - \frac{\delta}{\sqrt{n}})$ and $\hat{x} = x - \Delta x, \hat{y} = y - \Delta y, \hat{z} = z - \Delta z$, where $(\Delta x, \Delta y, \Delta z)$ is the solution of (3.11) with the parameter $\hat{\mu}$ and with $\bar{H}$. Then,*

- (a) $\|\hat{U}\hat{V}e - \hat{\mu}Pe\| \leq \theta\hat{\mu}$.
- (b) $\hat{x} \in \mathcal{S}^0, (\hat{y}, \hat{z}) \in \mathcal{T}^0$.
- (c) $g(\hat{x}, \hat{y}, \hat{z}) = \mathbf{E}(\hat{x}'\hat{z}) = n\hat{\mu}$.

Following the proof of Monteiro and Adler by using the above Euclidean norm, the parameters δ and θ should satisfy the following conditions:

$$(6.5) \quad 0 < \theta < \frac{\bar{p}}{2}, \quad \frac{\theta^2 + \delta^2}{2(\bar{p} - \theta)} \leq \theta \left(1 - \frac{\delta}{\sqrt{n}}\right),$$

which implies

$$(6.6) \quad \delta < \frac{\bar{p}}{\sqrt{2}} \leq \frac{1}{K\sqrt{2}}.$$

By the construction of the initial point, (x^0, y^0, z^0) satisfies the condition of Theorem 6.1. Theorem 6.1 is then the basis for the induction to prove that the sequence of points generated by Algorithm 3 satisfies (a), (b), and (c) of that theorem.

TABLE 6.1
Numerical results for Algorithm 2.

Problem	Stages	Scenarios	Deterministic equivalent	Optimal value	α	Initial gap	Iterations
CEP1	2	6	57×154	575035	0.6	1.05E+11	60
PGP2	2	9	38×189	465.22	0.6	2.35E+9	53
RG1	2	10	50×122	542.26	0.6	1.82E+9	53
RG2	2	200	410×1022	550.38	0.6	1.82E+9	53
RG3	2	300	3608×6628	133.22	0.667	6.89E+12	86
RG4	2	1000	12008×22028	162.06	0.667	6.89E+12	86
SC205	2	50	9613×14374	−50.25	0.667	6.78E+12	86
SC205	3	50	9173×13694	−46.66	0.667	6.78E+12	86
SCSD1	3	50	2070×21090	29.33	0.7	4.19E+11	89

COROLLARY 6.2. *The sequence of points (x^k, y^k, z^k) generated by Algorithm 3 satisfies, for all $k = 1, 2, \dots$*

- (a) $\|U^k V^k e - \mu_k P e\| \leq \theta \mu_k$.
- (b) $x^k \in \mathcal{S}^0, (y^k, z^k) \in \mathcal{T}^0$.
- (c) $g(x^k, y^k, z^k) = \mathbf{E}(x^{k'} z^k) = n \mu_k$.

In order to check the convergence of Algorithm 3, use Proposition 4.1 in [16] to obtain the following proposition.

PROPOSITION 6.3. *The total number of iterations performed by Algorithm 3 is no greater than*

$$\hat{k} = \left\lceil \log(n \epsilon^{-1} \mu_0) \frac{\sqrt{n}}{\delta} \right\rceil,$$

where $\epsilon > 0$ denotes the tolerance of the duality gap and $\mu_0 = \frac{\mathbf{E}(x^{0'} z^0)}{n}$.

Since $\mu_0 = 2^{6L}$,

$$\hat{k} \leq 1 + \log(n \epsilon^{-1} 2^{6L}) \frac{\sqrt{n}}{\delta} = 1 + (\log n - \log \epsilon + 6L) \frac{\sqrt{n}}{\delta}.$$

However, from (6.6), $\delta < \frac{1}{K\sqrt{2}}$. Therefore, $\hat{k}$ is of $O(\sqrt{n}K \log n)$, which is worse than $O(\sqrt{N} \log N)$. Nevertheless, if μ is updated by a constant factor $0 < \alpha < 1$, and the algorithm starts from a duality gap equal to $n\mu_0$ and terminates when

$$n\mu_0 \alpha^k < \epsilon,$$

it will need $O(\log n)$ iterations to terminate (for constants ϵ and L). On the other hand, if the algorithm starts from $N\mu_0$ as Algorithm 2, it will need $O(\log N)$ iterations to terminate. So, whenever an update by a constant factor of μ is possible, and it is possible in most cases, Algorithm 3 seems to be superior to Algorithm 2. However, if Algorithm 3 works efficiently with $\bar{\alpha}$, Algorithm 2 will use $\alpha < \bar{\alpha}$ in most cases. So Algorithm 3 may be superior only when $N \gg n$, i.e., when the number of variables in the deterministic equivalent program is much larger than the number of random decision variables in the stochastic program.

TABLE 6.2
Numerical results for Algorithm 3.

Problem	Stages	Scenarios	Deterministic equivalent	Optimal value	α	Initial gap	Iterations
CEP1	2	6	57×154	575035	0.667	$3.07\text{E}+10$	73
PGP2	2	9	38×189	465.22	0.85	$4.70\text{E}+8$	153
RG1	2	10	50×122	542.26	0.6	$3.62\text{E}+8$	49
RG2	2	200	410×1022	550.38	0.6	$3.62\text{E}+8$	49
RG3	2	300	3608×6628	133.22	0.7	$4.06\text{E}+10$	83
RG4	2	1000	12008×22028	162.06	0.7	$4.06\text{E}+10$	83
SC205	2	50	9613×14374	-50.25	0.667	$3.83\text{E}+11$	79
SC205	3	50	9173×13694	-46.66	0.667	$3.83\text{E}+11$	79
SCSD1	3	50	2070×21090	29.33	0.75	$9.34\text{E}+10$	105

7. Numerical results. Algorithms 2 and 3 were applied to some test problems, using MATLAB. The actual time for solving the largest test problem was reasonable, taking into account that MATLAB is relatively slow for that kind of computation. Problems CEP1, PGP2, SC205, and SCSD1 are known from the literature and can be found in [11], [12], and [14]. Problems SC205 and SCSD1 are deterministic problems that were modified to be stochastic programming problems. Problems RG1, RG2, RG3, and RG4 are randomly generated two-stage stochastic linear programs.

Tables 6.1 and 6.2 contain the results of applying Algorithms 2 and 3, respectively, to these test problems. In both tables, the number of scenarios and the size of the deterministic equivalent linear programs are given. The optimal value is accurate up to two digits after the decimal point. The updating factor for μ is α . The initial duality gap used by the algorithms and the number of iterations that were executed are given in Tables 6.1 and 6.2. Updating μ by a constant factor was possible for all the tested problems.

From Tables 6.1 and 6.2 it can be seen that both algorithms terminated with the same optimal values for all the test problems. Whenever Algorithm 2 could use $\bar{\alpha}$ as the updating factor of μ , Algorithm 3 could only use a larger updating factor $\alpha \geq \bar{\alpha}$. This is natural since Algorithm 2 is theoretically more efficient than Algorithm 3. However, although Algorithm 3 uses a larger updating factor, it starts with a smaller duality gap. In the case of updating μ by a constant factor, since the number of iterations is a function of solely α , the initial duality gap, and the accuracy, Algorithm 3 requires a smaller number of iterations than Algorithm 2, unless it uses a much larger updating factor α (see PGP2 and SCSD1).

When Algorithm 3 needs less iterations than Algorithm 2, the difference between the numbers of iterations required by the algorithms is not large. However, saving the execution of one or two iterations can save a significant amount of time when the number of scenarios is large.

Acknowledgments. The author is thankful to Professors Mordecai Avriel and Arkadi Nemirovski of the Faculty of Industrial Engineering and Management at the Technion for their constant help and support.

REFERENCES

- [1] E. M. L. BEALE, *On minimizing a convex function subject to linear inequalities*, J. Roy. Statist. Soc., Ser. B, 17 (1955), pp. 173–184.

- [2] E. M. L. BEALE, J. J. H. FORREST, AND C. J. TAYLOR, *Multi-time-period stochastic programming*, in Stochastic Programming, M. A. H. Dempster, ed., Academic Press, London, 1980, pp. 387–402.
- [3] A. J. BERGER, J. M. MULVEY, E. ROTHBERG, AND R. J. VANDERBEI, *Solving Multistage Stochastic Programs Using Tree Dissection*, Technical Report SOR 92-5, Department of Civil Engineering and Operations Research, Princeton University, Princeton, NJ, 1995.
- [4] J. R. BIRGE, *Decomposition and partitioning methods for multistage stochastic linear programs*, Oper. Res., 33 (1985), pp. 989–1007.
- [5] J. R. BIRGE AND D. HOLMES, *Efficient solution of two-stage stochastic linear programs using interior point methods*, Comput. Optim. Appl., 1 (1992), pp. 245–276.
- [6] J. CZYZYK, R. FOURER, AND S. MEHROTRA, *A Study of the Augmented System and Column-Splitting Approaches for Solving Two-Stage Stochastic Linear Programs by Interior-Point Methods*, Technical Report 93-05, Department of Industrial and Management Science, Northwestern University, Evanston, IL, 1993.
- [7] G. B. DANTZIG, *Linear programming under uncertainty*, Management Science, 1 (1955), pp. 197–206.
- [8] G. B. DANTZIG, *Linear Programming and Extensions*, Princeton University Press, Princeton NJ, 1963.
- [9] Y. ERMOLIEV AND R. J.-B. WETS, *Numerical Techniques for Stochastic Optimization*, Springer-Verlag, Berlin, 1988.
- [10] A. V. FIACCO AND G. P. MCCORMICK, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley and Sons, New York, 1968.
- [11] J. HIGLE, S. SEN, AND D. S. YAKOWITZ, *Finite Master Programs in Stochastic Decomposition*, Technical Report, Dept. of Systems and Industrial Engineering, The University of Arizona, Tucson, AZ, 1990.
- [12] J. K. HO AND E. LOUTE, *A set of staircase linear programming test problems*, Math. Programming, 20 (1981), pp. 245–250.
- [13] G. INFANGER, *Planning Under Uncertainty: Solving Large-Scale Stochastic Linear Programs*, Boyd and Fraser, Danvers, MA, 1994.
- [14] F. LOUVEAUX AND N. SMEERS, *Optimal investment for electricity generation: A stochastic model and a test problem*, in Numerical Techniques for Stochastic Optimization, Y. Ermoliev and R. J.-B. Wets, eds., Springer-Verlag, Berlin, 1988.
- [15] I. J. LUSTIG, J. M. MULVEY, AND T. J. CARPENTER, *Formulating stochastic two-stage programs for interior point methods*, Oper. Res., 39 (1991), pp. 757–770.
- [16] R. D. C. MONTEIRO AND I. ADLER, *Interior path following primal-dual algorithms. Part I: Linear programming*, Math. Programming, 44 (1988), pp. 24–41.
- [17] R. D. C. MONTEIRO AND I. ADLER, *Interior path following primal-dual algorithms. Part II: Convex quadratic programming*, Math. Programming, 44 (1989), pp. 43–66.
- [18] J. M. MULVEY AND A. RUSZCZYNSKI, *A diagonal quadratic approximation method for large scale linear programs*, Oper. Res. Lett., 12 (1992), pp. 205–215.
- [19] L. NAZARETH AND R. J.-B. WETS, *Nonlinear programming techniques applied to stochastic programs with recourse*, in Numerical Techniques for Stochastic Optimization, Y. Ermoliev and R. J.-B. Wets, eds., Springer-Verlag, Berlin, 1988, pp. 95–121.
- [20] Y. NESTEROV AND A. NEMIROVSKII, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, 1994.
- [21] A. RUSZCZYNSKI, *Parallel decomposition of multistage stochastic programming problems*, Math. Programming, 58 (1993), pp. 201–228.
- [22] E. SCHWEITZER AND M. AVRIEL, *A Gaussian upper bound for Gaussian multi-stage stochastic linear programs*, Math. Programming, 77 (1977), pp. 1–21.
- [23] R. M. VAN SLYKE AND R. WETS, *L-shaped linear programs with applications to optimal control and stochastic programming*, SIAM J. Appl. Math., 17 (1969), pp. 638–663.
- [24] R. J.-B. WETS, *Stochastic programs with fixed recourse: The equivalent deterministic program*, SIAM Rev., 16 (1974), pp. 309–339.
- [25] R. J.-B. WETS, *Stochastic programming*, in Handbook in Operations Research and Management Science 1, G. L. Nemhauser, A. H. G. Kan, and M. J. Todd, eds., Elsevier Science, Amsterdam, 1989, pp. 573–629.

APPROXIMATION SCHEMES FOR INFINITE LINEAR PROGRAMS*

ONÉSIMO HERNÁNDEZ-LERMA[†] AND JEAN B. LASSERRE[‡]

Abstract. This paper presents approximation schemes for an infinite linear program. In particular, it is shown that, under suitable assumptions, the program's optimum value can be approximated by the values of finite-dimensional linear programs, and that, in addition, every accumulation point of a sequence of optimal solutions for the approximating programs is an optimal solution for the original problem.

Key words. infinite-dimensional linear programming, aggregation-relaxation (of constraints), inner approximations

AMS subject classifications. 90C05, 90C48

PII. S1052623497315768

1. Introduction. The main objective of this paper is to introduce an approximation scheme for infinite linear programs using finite-dimensional programs. This is done in three parts. Given an infinite linear program—say (in notation explained in section 2),

$$\mathbb{P} : \text{minimize} \{ \langle x, c \rangle \mid Ax = b, x \in \mathcal{X}_+ \},$$

first we show that, under suitable assumptions, $\mathbb{P}$ is *solvable* (Theorem 2.1). Next we introduce an *aggregation-relaxation* scheme to approximate the constraint equation $Ax = b$ by a *finite* (and increasing) number of constraints. Finally, the latter scheme is complemented with a sequence of finite *inner approximations* of the variable x , which define the required finite-dimensional programs. The last two schemes are shown to “converge” (Theorems 3.1 and 3.2) when the underlying vector spaces are reasonably nice (Assumptions 3.1 and 3.2). We also present three examples (on the *mass-transfer problem*, *Markov control processes*, and *semi-infinite linear programs*) to illustrate how one can verify the required hypotheses.

Infinite (and semi-infinite) linear programming (LP) problems appear in many branches of mathematics, including probability theory, numerical analysis, optimal control, mathematical economics, and operations research, to name just a few—see [1, 4, 6, 7, 9, 10, 12, 13, 14, 15, 16, 18, 19, 21, 22]. Of course, by their very nature, these problems are in general difficult (if perhaps not impossible) to solve in “closed” form, and one is forced to use some kind of approximation scheme. There are many related works in this direction, but typically, as can be seen in the above references, these schemes are specifically designed for particular classes of problems. An exception is the paper by Vershik and Temel't [22] (see also [21]), where an approximation scheme for general linear programs is proposed. Our approach is similar in spirit to that in [22], but with important differences. For instance, they use the so-called strong Helly topology (introduced in game theory), which excludes many important cases—a

*Received by the editors January 29, 1997; accepted for publication (in revised form) September 2, 1997; published electronically August 26, 1998. This research was partially supported by the CNRS (France)–CONACYT (México) Scientific Cooperation Program. The work of the first author was also partially supported by CONACYT grant 3115P–E9608.

<http://www.siam.org/journals/siopt/8-4/31576.html>

[†]Departamento de Matemáticas, CINVESTAV-IPN, Apartado Postal 14–740, México D.F. 07000, México (ohernand@math.cinvestav.mx).

[‡]LAAS-CNRS, 7 Avenue du Colonel Roche, 31077 Toulouse Cédex, France (lasserre@laas.fr).

similar remark holds for [21], which uses normed topologies. In contrast, we work here with *weak* topologies, which allow us to handle a larger class of problems and which, on the other hand, give a simple and natural interpretation to our approximation scheme (see the paragraph following (3.1)). Moreover, we prove the convergence of the sequence (and of subsequences—see Theorems 3.1, 3.2, and 3.3) of approximating optimal values, thus filling an important gap in the results of [22].

The remainder of the paper is organized as follows. In section 2 we specify the linear program we shall be dealing with and give conditions under which it is solvable. Our approximation results are presented in section 3, and, finally, we conclude in section 4 with some relevant remarks and examples.

2. Preliminaries. For ease of reference, we shall essentially follow the notation of Anderson and Nash [1, section 3.3]. (Similar material is presented in [9, section 6.2] and [14, section 4.2].)

Let $(\mathcal{X}, \mathcal{X}')$ and $(\mathcal{Y}, \mathcal{Y}')$ be two dual pairs of vector spaces. The spaces $\mathcal{X}$ and $\mathcal{X}'$ are assumed to be endowed with the weak topologies $\sigma(\mathcal{X}, \mathcal{X}')$ and $\sigma(\mathcal{X}', \mathcal{X})$, respectively, unless some other topology is explicitly used, and similarly for $\mathcal{Y}$ and $\mathcal{Y}'$. Let $A : \mathcal{X} \rightarrow \mathcal{Y}$ be a *weakly continuous* linear map with adjoint $A' : \mathcal{Y}' \rightarrow \mathcal{X}'$, i.e.,

$$\langle Ax, y' \rangle = \langle x, A'y' \rangle \quad \forall x \in \mathcal{X}, y' \in \mathcal{Y}'.$$

We denote by $\mathcal{X}_+$ the positive cone in $\mathcal{X}$ and by $\mathcal{X}'_+$ the *dual cone* in $\mathcal{X}'$, i.e.,

$$\mathcal{X}'_+ := \{x' \in \mathcal{X}' \mid \langle x, x' \rangle \geq 0 \quad \forall x \in \mathcal{X}_+\}.$$

Let $c \in \mathcal{X}'$ and $b \in \mathcal{Y}$ be given vectors and consider the (“equality constrained”) primal linear program

$$\mathbb{P} : \text{minimize} \quad \langle x, c \rangle$$

$$(2.1) \quad \text{subject to} \quad Ax = b, \quad x \in \mathcal{X}_+,$$

and its *dual*

$$\mathbb{P}^* : \text{maximize} \quad \langle b, y' \rangle$$

$$(2.2) \quad \text{subject to} \quad c - A'y' \in \mathcal{X}'_+, \quad y' \in \mathcal{Y}'.$$

A vector $x \in \mathcal{X}$ is said to be a *feasible solution* for $\mathbb{P}$ if it satisfies (2.1), and $\mathbb{P}$ is called *consistent* if it has a feasible solution. If $\mathbb{P}$ is consistent, then its (optimum) *value* is defined as $\inf \mathbb{P} := \inf \{\langle x, c \rangle \mid x \text{ is feasible for } \mathbb{P}\}$. It is said that $\mathbb{P}$ is *solvable* if there is a feasible solution x^* that achieves the optimum value. In this case, x^* is called an *optimal solution* for $\mathbb{P}$, and the value $\inf \mathbb{P}$ is then written as $\min \mathbb{P} = \langle x^*, c \rangle$. For the dual problem $\mathbb{P}^*$ we use a similar notation—in particular, we write its value as $\sup \mathbb{P}^*$, which is replaced by $\max \mathbb{P}^*$ if $\mathbb{P}^*$ is solvable. Finally, if $\inf \mathbb{P} = \sup \mathbb{P}^*$, it is said that there is *no duality gap*.

We next state an assumption that is supposed to hold throughout the following. It ensures, in particular, that $\mathbb{P}$ is solvable (see Theorem 2.1).

Assumption 2.1. (a) $\mathbb{P}^*$ is consistent.

(b) $\mathbb{P}$ has a feasible (nonoptimal) solution x_0 with $\langle x_0, c \rangle > 0$ and, moreover, the set

$$\mathcal{X}^0 := \{x \in \mathcal{X}_+ | \langle x, c \rangle \leq \langle x_0, c \rangle\}$$

is weakly sequentially compact.

Speaking of assumptions, recall that the linear map A is required to be weakly continuous (equivalently, A' maps $\mathcal{Y}'$ into $\mathcal{X}'$; see [1, p. 37, Prop. 4], [14, Prop. 4.3], or [17, p. 38, Prop. 12]). We also note the following.

Remark 2.1. The set $\mathcal{X}^0$ is not compact if $\langle x, c \rangle < 0$ for some $x \in \mathcal{X}_+$. Thus, Assumption 2.1 implies that the cost vector c is *nonnegative*; that is, c is in $\mathcal{X}'_+$. This in turn implies that the set $\mathcal{X}_a := \{x \in \mathcal{X}_+ | \langle x, c \rangle \leq a\}$ is weakly sequentially compact for every $a \in \mathbb{R}_+$, since $\mathcal{X}_a = (a/\langle x_0, c \rangle)\mathcal{X}^0$.

THEOREM 2.1. *Suppose that Assumption 2.1 holds. Then*

(a) $\mathbb{P}$ is solvable.

(b) There is no duality gap, i.e., (using (a)),

$$(2.3) \quad \min \mathbb{P} = \sup \mathbb{P}^*.$$

Proof. (a) As $\mathbb{P}$ and $\mathbb{P}^*$ are both assumed to be consistent, the *weak duality* property holds [1, Thm. 3.1]; that is, the values $\inf \mathbb{P}$ and $\sup \mathbb{P}^*$ are both finite, and $\inf \mathbb{P} \geq \sup \mathbb{P}^*$. Now let x_0 be as in Assumption 2.1(b), and let $\{x_n\}$ be a *minimizing sequence* for $\mathbb{P}$; that is, each $x_n \in \mathcal{X}_+$ is feasible for $\mathbb{P}$ and $\langle x_n, c \rangle \downarrow \inf \mathbb{P}$. Then, for all n sufficiently large, the vector x_n is in the weakly sequentially compact set $\mathcal{X}^0$, and, therefore, there is a subsequence $\{x_m\}$ of $\{x_n\}$ and a vector $x^* \in \mathcal{X}_+$ such that $x_m \rightarrow x^*$ in the weak topology $\sigma(\mathcal{X}, \mathcal{X}')$, i.e.,

$$\langle x_m, x' \rangle \rightarrow \langle x^*, x' \rangle \quad \forall x' \in \mathcal{X}'.$$

This implies that x^* is feasible for $\mathbb{P}$, since

$$(2.4) \quad \langle Ax^*, x' \rangle = \langle x^*, A'x' \rangle = \lim \langle x_m, A'x' \rangle = \lim \langle Ax_m, x' \rangle = \langle b, x' \rangle$$

$\forall x' \in \mathcal{X}'$, and it also yields that x^* is an optimal solution for $\mathbb{P}$, as (taking $x' = c$)

$$\langle x^*, c \rangle = \lim \langle x_m, c \rangle = \inf \mathbb{P}.$$

(b) Consider the subset $D := \{(Ax, \langle x, c \rangle) | x \in \mathcal{X}_+\}$ of $\mathcal{Y} \times \mathbb{R}$. Then, according to Theorem 3.10 in [1, p. 53], to prove (2.3) it suffices to show that

$$(2.5) \quad D \text{ is weakly closed.}$$

To prove this, let $(N, \leq)$ be a directed set, and let $\{x_\alpha, \alpha \in N\}$ be a net in $\mathcal{X}_+$ such that $(Ax_\alpha, \langle x_\alpha, c \rangle)$ converges weakly to some (y, ρ) in $\mathcal{Y} \times \mathbb{R}$, i.e.,

$$(2.6) \quad \text{(i) } \langle Ax_\alpha, y' \rangle \rightarrow \langle y, y' \rangle \quad \forall y' \in \mathcal{Y}' \quad \text{and} \quad \text{(ii) } \langle x_\alpha, c \rangle \rightarrow \rho.$$

We wish to show that (y, ρ) is in D ; that is, there exists x_* in $\mathcal{X}_+$ such that

$$(2.7) \quad \text{(i) } Ax_* = y, \quad \text{and} \quad \text{(ii) } \langle x_*, c \rangle = \rho.$$

Now, for any given $\varepsilon > 0$, the convergence in (2.6)(ii) implies the existence of α_0 in N such that $\langle x_\alpha, c \rangle \leq a$, with $a := \rho + \varepsilon$, $\forall \alpha \geq \alpha_0$, and $a > 0$ since $\rho \geq 0$. Hence,

the set $\mathcal{X}_a$ in Remark 2.1 following Assumption 2.1 is weakly sequentially compact, and, therefore, there is a sequence $\{\alpha(n)\}$ in N and a vector x_* in $\mathcal{X}_+$ such that $x_{\alpha(n)}$ converges weakly to x_* , i.e.,

$$(2.8) \quad \langle x_{\alpha(n)}, x' \rangle \rightarrow \langle x_*, x' \rangle \quad \forall x' \in \mathcal{X}'.$$

In particular, taking $x' = c$, we obtain (2.7)(ii), whereas (2.7)(i) follows from (2.8) and (2.6)(i) (by a calculation similar to (2.4)). This completes the proof of (2.5). $\square$

Before considering the approximation procedures we are interested in, we should note that the most restrictive part of Assumption 2.1 is the requirement on the set $\mathcal{X}^0$. However, in section 4 we show that it holds in many cases by an appropriate choice of spaces. Concerning the consistency of $\mathbb{P}$, necessary and sufficient conditions for the existence of solutions to (2.1) can be obtained from variants of *Farkas' theorem*, as in [5, 11], for example.

To motivate the approximation schemes introduced in section 3, let $Y' \subset \mathcal{Y}'$ and $X_+ \subset \mathcal{X}_+$ be given subsets of $\mathcal{Y}'$ and $\mathcal{X}_+$. Then the program

$$(2.9) \quad \begin{aligned} &\mathbb{P}(Y') : \text{minimize } \langle x, c \rangle \\ &\text{subject to } \langle Ax - b, y' \rangle = 0 \quad \forall y' \in Y', \\ &x \in \mathcal{X}_+, \end{aligned}$$

is called an *aggregation* (of constraints) of $\mathbb{P}$, and the program

$$(2.10) \quad \begin{aligned} &\mathbb{P}(X_+) : \text{minimize } \langle x, c \rangle \\ &\text{subject to } Ax = b, \quad x \in X_+, \end{aligned}$$

is called an *inner approximation* of $\mathbb{P}$. If x satisfies (2.10), then it also satisfies (2.1); similarly, if x satisfies (2.1), then it obviously satisfies (2.9). Hence,

$$\inf \mathbb{P}(Y') \leq \min \mathbb{P} \leq \inf \mathbb{P}(X_+).$$

Moreover, we have the following proposition.

PROPOSITION 2.2. (a) *If $\mathbb{P}$ is solvable and $Y' \subset \mathcal{Y}'$ is weakly dense in $\mathcal{Y}'$, then $\mathbb{P}(Y')$ is equivalent to $\mathbb{P}$ in the sense that $\mathbb{P}(Y')$ is also solvable and*

$$(2.11) \quad \min \mathbb{P}(Y') = \min \mathbb{P}.$$

(b) *If $\mathbb{P}$ is solvable and $X_+ \subset \mathcal{X}_+$ is dense in $\mathcal{X}_+$, then there is a sequence $\{x_n\}$ in X_+ such that*

$$(2.12) \quad \langle x_n, c \rangle \rightarrow \min \mathbb{P}.$$

Proof. (a) By definition of dual pair and the denseness of Y' , the statements (2.1) and (2.9) are equivalent, i.e.,

$$(2.13) \quad \begin{aligned} Ax = b &\Leftrightarrow \langle Ax - b, y' \rangle = 0 \quad \forall y' \in \mathcal{Y}' \\ &\Leftrightarrow \langle Ax - b, y' \rangle = 0 \quad \forall y' \in Y'. \end{aligned}$$

This implies (a) because if there exists a feasible solution x for $\mathbb{P}(Y')$ with

$$(2.14) \quad \langle x, c \rangle < \min \mathbb{P},$$

then, by (2.13), x is a feasible solution for $\mathbb{P}$ and (2.14) would contradict the optimality of $\mathbb{P}$.

(b) If x is an optimal solution of $\mathbb{P}$, then by the denseness of X_+ , there is a sequence $\{x_n\}$ in X_+ that converges to x in the weak topology $\sigma(\mathcal{X}, \mathcal{X}')$, i.e.,

$$\langle x_n, x' \rangle \rightarrow \langle x, x' \rangle \quad \forall x' \in \mathcal{X}.$$

In particular, taking $x' = c$ we obtain (2.12). $\square$

Proposition 2.2, although quite elementary, provided the initial idea for the approximation schemes to be introduced next.

3. The approximation scheme. In addition to Assumption 2.1, our approximation results require two additional hypotheses. The first one, Assumption 3.1 below, allows us to approximate the constraint equation $Ax = b$ by a *finite* (and increasing) *number of constraints* (see Theorem 3.1). The second, Assumption 3.2, allows us to approximate a feasible solution for $\mathbb{P}$ by linear combinations of a *finite number of vectors in $\mathcal{X}$* (see Theorem 3.2). We shall use the following notation.

Notation. Let E be a subset of a vector space. Then $sp(E)$ denotes the *subspace spanned* (or generated) *by E* , and $coc(E)$ stands for the *convex cone generated by E* (that is, $coc(E)$ is the cone generated by the convex hull of E).

Assumption 3.1. There is an increasing sequence of finite sets $\mathcal{Y}'_k$ in $\mathcal{Y}'$ such that $\mathcal{Y}'_\infty := \cup_{k=1}^\infty sp(\mathcal{Y}'_k)$ is weakly dense in $\mathcal{Y}'$.

Assumption 3.2. There is an increasing sequence of finite sets $\mathcal{X}_n$ in $\mathcal{X}_+$ such that $\mathcal{X}_+^\infty := \cup_{n=1}^\infty coc(\mathcal{X}_n)$ is weakly dense in $\mathcal{X}_+$.

Note that the “denseness” properties required in Assumptions 3.1 and 3.2 are with respect to *weak* topologies. This permits us to handle some problems (for example, in measure spaces) in which a strong denseness hypothesis might not be possible or it would be too restrictive.

3.1. An aggregation-relaxation scheme. For every $k = 1, 2, \dots$, let $\mathcal{Y}'_k \subset \mathcal{Y}'$ be as in Assumption 3.1, and let $\{\varepsilon_k\}$ be a given sequence of positive numbers such that $\varepsilon_k \downarrow 0$. Instead of the original primal program $\mathbb{P}$ consider the linear program

$$(3.1) \quad \begin{aligned} &\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k) : \quad \text{minimize } \langle x, c \rangle \\ &\text{subject to } |\langle Ax - b, y' \rangle| \leq \varepsilon_k \quad \forall y' \in \mathcal{Y}'_k, \quad x \in \mathcal{X}_+. \end{aligned}$$

We can view $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$ as a two-step approximation. First we “aggregate” the constraints $Ax = b$ into a finite number of constraints (cf. (2.9)), say,

$$(3.2) \quad \langle Ax, y'_i \rangle = \langle b, y'_i \rangle, \quad i = 1, \dots, |\mathcal{Y}'_k|,$$

where y'_i is in $\mathcal{Y}'_k$ for $i = 1, \dots, |\mathcal{Y}'_k| := \text{cardinality of } \mathcal{Y}'_k$; then we “relax” the equality (3.2) to the inequality in (3.1). (Observe that the aggregation step corresponds to a row-type constraint aggregation commonly used in finite-dimensional LP.) Moreover, the constraints $|\langle Ax - b, y' \rangle| \leq \varepsilon_k$, $y' \in \mathcal{Y}'_k$, can be interpreted as saying that Ax is in $N(b, \mathcal{Y}'_k, \varepsilon_k)$, where $N(b, \mathcal{Y}'_k, \varepsilon_k)$ is a *weak $\sigma(\mathcal{Y}, \mathcal{Y}')$ -neighborhood* of $b \in \mathcal{Y}$ (see,

for instance, [17, p. 32]). This shows that the weak topology $\sigma(\mathcal{Y}, \mathcal{Y}')$ is indeed the appropriate topology to be used in Assumption 3.1. On the other hand, if we write b as $b = Ax_b$ for some $x_b \in X$ (for example, take x_b as the feasible solution x_0 in Assumption 2.1(b)), then the constraint

$$|\langle Ax - Ax_b, y' \rangle| \leq \varepsilon_k, \quad y' \in \mathcal{Y}'_k,$$

means that x is in $N(x_b, A, \mathcal{Y}'_k, \varepsilon_k)$, a neighborhood of $x_b \in \mathcal{X}$ in the weak Helly topology on $\mathcal{X}$ induced by the linear map A (see [22]).

THEOREM 3.1. *Suppose that Assumptions 2.1 and 3.1 hold. Then*

(a) $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$ *is solvable for every* $k = 1, 2, \dots$

(b) *For every* $k = 1, 2, \dots$, *let* x_k *be an optimal solution for* $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$. *Then, as* $k \rightarrow \infty$,

$$(3.3) \quad \langle x_k, c \rangle \uparrow \min \mathbb{P},$$

and, further, every weak accumulation point of $\{x_k\}$ *is an optimal solution for* $\mathbb{P}$.

Proof. (a) By Theorem 2.1(a), $\mathbb{P}$ is solvable. In addition, every feasible solution for $\mathbb{P}$ is also a feasible solution for $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$, so that

$$(3.4) \quad \inf \mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k) \leq \min \mathbb{P} \quad (k = 1, 2, \dots),$$

and, on the other hand, the value sequence $\inf \mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$ increases with k . To show that $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$ is solvable, we proceed as in the proof of Theorem 2.1(a).

Consider a minimizing sequence $\{x_n\}$ for $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$. By the same arguments used in the proof of Theorem 2.1(a), there is $x \in \mathcal{X}_+$ and a subsequence $\{x_m\}$ of $\{x_n\}$ such that $x_m \rightarrow x$ in the weak topology $\sigma(\mathcal{X}, \mathcal{X}')$. Moreover (as noted above), the condition

$$|\langle Ax_m - b, y' \rangle| \leq \varepsilon_k \quad \forall y' \in \mathcal{Y}'_k$$

means that Ax_m is in the weak neighborhood $N(b, \mathcal{Y}'_k, \varepsilon_k)$ of b . Thus, as $x_m \rightarrow x$ weakly and A is weakly continuous, the vector Ax is also in $N(b, \mathcal{Y}'_k, \varepsilon_k)$, i.e.,

$$|\langle Ax - b, y' \rangle| \leq \varepsilon_k \quad \forall y' \in \mathcal{Y}'_k.$$

Therefore, x is feasible—and hence optimal (see the proof of Theorem 2.1(a))—for $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$.

(b) Let x_k ($k = 1, 2, \dots$) be an optimal solution for $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$, i.e.,

$$\langle x_k, c \rangle = \min \mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k).$$

Then, by (3.4) and the monotonicity of the value sequence $\langle x_k, c \rangle$, there is a number $s^* \leq \min \mathbb{P}$ such that $\langle x_k, c \rangle \uparrow s^*$. Using again a weak sequential compactness argument, there is a vector x^* in $\mathcal{X}_+$ and a subsequence $\{x_l\}$ of $\{x_k\}$ such that $x_l \rightarrow x^*$ weakly; in particular, as c is in $\mathcal{X}'$,

$$\langle x_l, c \rangle \rightarrow \langle x^*, c \rangle = s^* \leq \min \mathbb{P}.$$

The latter inequality—namely, $\langle x^*, c \rangle \leq \min \mathbb{P}$ —implies that to complete the proof of part (b) it suffices to show that x^* is a feasible solution for $\mathbb{P}$, i.e.,

$$(3.5) \quad Ax^* = b,$$

because then, by definition of optimality, we would have the reverse inequality, $\langle x^*, c \rangle \geq \min \mathbb{P}$. On the other hand, to prove (3.5) it suffices to show that

$$(3.6) \quad \langle Ax^*, y' \rangle = \langle b, y' \rangle \quad \forall y' \text{ in } \mathcal{Y}'_\infty := \cup_{k=1}^\infty sp(\mathcal{Y}'_k),$$

since then Assumption 3.1 would yield that (3.6) holds in fact for all y' in $\mathcal{Y}'$.

Now, to prove (3.6), let y' be an arbitrary vector in $\mathcal{Y}'_\infty$ and write

$$(3.7) \quad \langle Ax^*, y' \rangle = \langle Ax^* - Ax_l, y' \rangle + \langle Ax_l - b, y' \rangle + \langle b, y' \rangle.$$

The first term on the right-hand side of (3.7) converges to zero as $l \rightarrow \infty$, since the weak convergence $x_l \rightarrow x^*$ and the weak continuity of A yield that, for all y' in $\mathcal{Y}'$,

$$(3.8) \quad \langle Ax_l, y' \rangle = \langle x_l, A'y' \rangle \rightarrow \langle x^*, A'y' \rangle = \langle Ax^*, y' \rangle.$$

On the other hand, as y' is in $\mathcal{Y}'_\infty$, there is a k such that y' is in $sp(\mathcal{Y}'_k)$, i.e.,

$$y' = \sum_{i=1}^{i(k)} \lambda_i y'_i \quad \text{with} \quad y'_i \in \mathcal{Y}'_k \text{ and } \lambda_i \in \mathbb{R} \ (i = 1, \dots, i(k)).$$

Hence, the term $\langle Ax_l - b, y' \rangle$ in (3.7) satisfies

$$(3.9) \quad |\langle Ax_l - b, y' \rangle| \leq \sum_{i=1}^{i(k)} |\lambda_i| |\langle Ax_l - b, y'_i \rangle|.$$

Moreover, as the sequence $\{\mathcal{Y}'_k\}$ is increasing, there is an $l(y')$ such that y'_i is in $\mathcal{Y}'_l$ for all $i = 1, \dots, i(k)$ and $l \geq l(y')$; thus, by (3.1),

$$|\langle Ax_l - b, y'_i \rangle| \leq \varepsilon_l \quad \forall i = 1, \dots, i(k) \text{ and } l \geq l(y').$$

Hence, by (3.9),

$$|\langle Ax_l - b, y' \rangle| \leq \varepsilon_l \sum_{i=1}^{i(k)} |\lambda_i| \rightarrow 0 \quad \text{as } l \rightarrow \infty.$$

To conclude, letting $l \rightarrow \infty$ in (3.7) we obtain (3.6), and (3.5) follows. $\square$

3.2. The inner approximation. Even if $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$ has only a *finite* number of constraints, it is still of course an infinite linear program. To obtain a “truly finite” approximation of $\mathbb{P}$ we shall next use Assumption 3.2 to approximate the vectors x in (3.1) by vectors $\sum_{i=1}^{n^*} \lambda_i x_i$ in $coc(\mathcal{X}_n)$, where n^* stands for the cardinality $|\mathcal{X}_n|$ of $\mathcal{X}_n$.

For every $n = 1, 2, \dots$, let $\mathcal{X}_n = \{x_1, \dots, x_{n^*}\}$ be as in Assumption 3.2 and consider the finite linear program

$$(3.10) \quad \begin{aligned} & \mathbb{P}(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k) : \text{minimize} \quad \sum_{i=1}^{n^*} \lambda_i \langle x_i, c \rangle \\ & \text{subject to} \quad \left| \left\langle b - \sum_{i=1}^{n^*} \lambda_i A x_i, y' \right\rangle \right| \leq \varepsilon_k \quad \forall y' \in \mathcal{Y}'_k, \\ & \quad \lambda_i \geq 0 \quad \forall i = 1, \dots, n^* := |\mathcal{X}_n|. \end{aligned}$$

THEOREM 3.2. *Suppose that Assumptions 2.1, 3.1, and 3.2 hold. Then for every k there exists $n(k)$ such that*

(a) For each $n \geq n(k)$, $\mathbb{P}(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k)$ is solvable and

$$(3.11) \quad \min \mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k) \leq \min \mathbb{P}(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k) \leq \min \mathbb{P} + \varepsilon_k.$$

(b) If $\{n_k\}$ is a sequence such that $n_k \geq n(k)$ for every $k = 1, 2, \dots$ and x_k is optimal for $\mathbb{P}(\mathcal{X}_{n_k}, \mathcal{Y}'_k, \varepsilon_k)$, then

$$(3.12) \quad \langle x_k, c \rangle = \min \mathbb{P}(\mathcal{X}_{n_k}, \mathcal{Y}'_k, \varepsilon_k) \rightarrow \min \mathbb{P} \text{ as } k \rightarrow \infty,$$

and, in addition, every weak accumulation point of $\{x_k, k = 1, 2, \dots\}$ is an optimal solution for $\mathbb{P}$.

Proof. (a) For any λ_i and x_i as in (3.10), the vector $x := \sum_{i=1}^{n^*} \lambda_i x_i$ satisfies (3.1). Thus, as $\mathbb{P}(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k)$ is a finite linear program bounded from below, it is solvable and its value satisfies

$$(3.13) \quad \min \mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k) \leq \min \mathbb{P}(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k) \quad \forall n, k.$$

To prove the second inequality in (3.11), fix an arbitrary k and let $x \in \mathcal{X}_+$ be an optimal solution for $\mathbb{P}$ (by Theorem 2.1(a), such an x exists). Then, by Assumption 3.2, there is a sequence $\{x_m\}$ in $\mathcal{X}_+^\infty$ such that $x_m \rightarrow x$ in the weak topology $\sigma(\mathcal{X}, \mathcal{X}')$. This implies two things. On the one hand, $\langle x_m, c \rangle \rightarrow \langle x, c \rangle = \min \mathbb{P}$, and, therefore, there exists $m_1(k)$ such that

$$(3.14) \quad \langle x_m, c \rangle \leq \min \mathbb{P} + \varepsilon_k \quad \forall m \geq m_1(k).$$

On the other, as $Ax = b$ and A is weakly continuous, there exists $m_2(k)$ such that

$$(3.15) \quad Ax_m \text{ is in } N(b, \mathcal{Y}'_k, \varepsilon_k) \quad \forall m \geq m_2(k),$$

where $N(b, \mathcal{Y}'_k, \varepsilon_k)$ is the weak neighborhood of b determined by (3.1).

Finally, note that, by definition of $\mathcal{X}_+^\infty$, every x_m is in $\text{coc}(\mathcal{X}_n)$ for some n , which combined with (3.15) yields that, for $m \geq m_2(k)$, x_m is feasible for $\mathbb{P}(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k)$, and this in turn, combined with (3.14) and the fact that the sets $\text{coc}(\mathcal{X}_n)$ form an increasing sequence, gives that

$$(3.16) \quad \min \mathbb{P}(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k) \leq \min \mathbb{P} + \varepsilon_k \quad \forall n \geq n(k),$$

where $n(k) := \max[m_1(k), m_2(k)]$.

(b) (3.12) follows from (3.11) and (3.3). To prove the last statement in (b), let $a_k := \min \mathbb{P} + \varepsilon_k$ and note that $a_k > 0$ and, by (3.16),

$$\langle x_k, c \rangle \leq a_k < a_1 \quad \forall k.$$

Thus, by the weak sequential compactness of $\mathcal{X}_{a_1}$ (see the Remark after Assumption 2.1), there exists x^* in $\mathcal{X}_+$ and a subsequence $\{x_{k_i}\}$ of $\{x_k\}$ such that $x_{k_i} \rightarrow x^*$ weakly. Finally, the proof is completed exactly as the last part of the proof of Theorem 3.1(b), replacing x_l with x_{n_i} in (3.7)–(3.9). $\square$

Remark 3.1. On the approximation of $\inf \mathbb{P}$. Theorems 3.1 and 3.2 give conditions, in particular, for

- (i) the approximation of the optimum value $\inf \mathbb{P}$ (see (3.3), (3.12)), and
- (ii) the existence of (sub)sequences which converge weakly to an optimal solution for $\mathbb{P}$.

If we are interested only in a result of type (i), then in Theorems 3.1 and 3.2 we may replace Assumption 2.1 with the following.

Assumption 2.1'. (a) $\mathbb{P}^*$ is consistent with a finite value, i.e., $\sup \mathbb{P}^* < \infty$.

(b) There is no duality gap, i.e., $\inf \mathbb{P} = \sup \mathbb{P}^*$.

In this case, the result corresponding to Theorem 3.1 (and similarly for Theorem 3.2 with the obvious changes) is as follows.

THEOREM 3.3. *Suppose that Assumptions 2.1' and 3.1 hold. Further, for every $k = 1, 2, \dots$, let x_k be a feasible solution for $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$ such that*

$$(3.17) \quad \langle x_k, c \rangle \leq \inf \mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k) + \varepsilon_k.$$

Then

$$(3.18) \quad \langle x_k, c \rangle \rightarrow \inf \mathbb{P} \quad \text{as } k \rightarrow \infty.$$

(Note that $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$ needs not be solved “exactly.”)

Proof. Assumption 2.1' ensures (by definition of *subvalue* of $\mathbb{P}$ and Theorem 3.1 in [1, p. 40]) that *any* sequence $\{x_k\}$ in $\mathcal{X}_+$ such that $Ax_k \rightarrow b$ weakly satisfies

$$(3.19) \quad \liminf_{k \rightarrow \infty} \langle x_k, c \rangle \geq \inf \mathbb{P} (= \sup \mathbb{P}^*).$$

In particular, by (3.1), the latter holds for the sequence $\{x_k\}$ in (3.17). On the other hand, as in (3.4), we obviously have

$$\inf \mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k) \leq \inf \mathbb{P},$$

so that, by (3.17) again,

$$\langle x_k, c \rangle \leq \inf \mathbb{P} + \varepsilon_k.$$

Hence,

$$\limsup_{k \rightarrow \infty} \langle x_k, c \rangle \leq \inf \mathbb{P},$$

which combined with (3.19) yields (3.18). $\square$

3.3. Penalization. To conclude this section we remark that there is a “penalization” variant of $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$ and $\mathbb{P}(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k)$. Namely, instead of the former program (see (3.1)) we may consider the following:

$$\widehat{\mathbb{P}}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k) : \text{minimize } [\langle x, c \rangle + \varepsilon_k^{-1} z]$$

$$\begin{aligned} \text{subject to } & \langle Ax - b, y' \rangle - z \leq 0, \quad y' \in \mathcal{Y}'_k, \\ & -\langle Ax - b, y' \rangle - z \leq 0, \quad y' \in \mathcal{Y}'_k, \\ \text{with } & x \in \mathcal{X}_+, \quad z \in \mathbb{R}_+. \end{aligned}$$

This is equivalent to

$$\min_{x \in \mathcal{X}_+} \left[\langle x, c \rangle + \varepsilon_k^{-1} \max_{y' \in \mathcal{Y}'_k} |\langle Ax - b, y' \rangle| \right],$$

so that the constraint $\langle Ax - b, y' \rangle = 0$ is “penalized” with ε_k^{-1} in the objective function. Then Theorem 3.1 remains valid (with virtually the same proof) when $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$ is replaced with $\widehat{\mathbb{P}}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$. Similarly, in Theorem 3.2 we may replace $\mathbb{P}(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k)$ with

$$\begin{aligned} \widehat{\mathbb{P}}(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k) : & \text{minimize } \left[\sum_{i=1}^{n^*} \lambda_i \langle x_i, c \rangle + \varepsilon_k^{-1} z \right] \\ \text{subject to } & \left\langle b - \sum_{i=1}^{n^*} \lambda_i A x_i, y' \right\rangle - z \leq 0, \quad y' \in \mathcal{Y}'_k, \\ & - \left\langle b - \sum_{i=1}^{n^*} \lambda_i A x_i, y' \right\rangle - z \leq 0, \quad y' \in \mathcal{Y}'_k, \\ & \text{with } z, \lambda_i \in \mathbb{R}_+, \quad i = 1, \dots, n^*. \end{aligned}$$

4. Remarks and examples. In this final section we make some general comments on our assumptions and present some examples.

Remark 4.1. Notation. We use below the following notation. Let $(X, \mathcal{B}(X))$ be a measurable space consisting of a metric space X and its Borel σ -algebra $\mathcal{B}(X)$. Then $M(X)$ denotes the Banach space of finite signed measures on $\mathcal{B}(X)$, with the total variation norm. Moreover,

$$M(X)_+ := \{\mu \in M(X) \mid \mu \geq 0\} \quad \text{and} \quad M_1(X) := \{\mu \in M(X)_+ \mid \mu(X) = 1\}$$

stand for the positive cone in $M(X)$ and the family of probability measures, respectively. In addition, $C_b(X)$ denotes the Banach space of continuous bounded functions on X , with the supremum norm, and $C_0(X)$ is the subspace of continuous functions vanishing at infinity. These two spaces coincide if X is *compact*, in which case we write $C_b(X) = C_0(X)$ as $C(X)$. Finally, we define the “inner product” $\langle \mu, f \rangle := \int f d\mu$ for μ in $M(X)$ and f in $C_b(X)$.

Remark 4.2. On Assumption 2.1(b). As was already mentioned, the weak sequential compactness of the set $\mathcal{X}^0$ in Assumption 2.1(b) is perhaps the most restrictive of our hypotheses. There are, however, many situations in which it holds, a trivial one being when the positive cone $\mathcal{X}_+$ or the whole space $\mathcal{X}$ itself is weakly sequentially compact; for a closely related case see Example 4.1. Another important situation is the following.

Suppose that $\mathcal{Z} \subset \mathcal{X}'$ is a *separable* Banach space and that $\mathcal{X}$ is the topological dual of $\mathcal{Z}$, so that the weak topology $\sigma(\mathcal{X}, \mathcal{Z})$ is in fact the so-called *weak* topology*. (This is the case, for example, if $\mathcal{X} := M(X)$ and $\mathcal{Z} := C_0(X)$ are as in Remark 4.1, with X a *locally compact separable metric space*, and, say, $\mathcal{X}' := B(X)$ is the space of bounded measurable functions on X .) Then, by the *Alaoglu* (or *Banach–Alaoglu–Bourbaki*) *theorem* [20], the weak* topology of a closed sphere in $\mathcal{X}$, i.e., $\{x \in \mathcal{X} \mid \|x\| \leq m\}$ for some $m \in \mathbb{R}_+$, is compact metrizable, and, hence, sequentially compact. If, in addition, A' maps $\mathcal{Y}'$ into $\mathcal{Z}$ (instead of into $\mathcal{X}'$), then we may replace Assumption 2.1(b) with “ $\mathcal{X}^0$ is $\sigma(\mathcal{X}, \mathcal{Z})$ -sequentially compact.” In turn, a sufficient condition for the latter property on $\mathcal{X}^0$ is, for instance, that for some a and m in $\mathbb{R}_+$: $x \in \mathcal{X}_+$ and $\langle x, c \rangle \leq a \Rightarrow \|x\| \leq m$. (The latter holds in the above-mentioned example if c is a positive function, bounded away from zero, either in $\mathcal{Z} := C_b(X)$ or in $\mathcal{X}' := B(X)$.)

Remark 4.3. On the dual programs. Let us consider the duals of $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$ and $\mathbb{P}(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k)$ (cf. (2.2)). With $\mathcal{Y}'_k := \{y'_1, \dots, y'_{k^*}\}$, where $k^* := |\mathcal{Y}'_k|$, the former reads

$$(4.1) \quad \begin{aligned} \mathbb{P}^*(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k) : \text{maximize} \quad & \left[\left\langle b, \sum_{i=1}^{k^*} (\gamma_i - \nu_i) y'_i \right\rangle + \varepsilon_k \sum_{i=1}^{k^*} (\gamma_i + \nu_i) \right] \\ \text{subject to} \quad & c - A' \sum_{i=1}^{k^*} (\gamma_i - \nu_i) y'_i \in \mathcal{X}'_+, \quad \gamma_i, \nu_i \in \mathbb{R}_+, \end{aligned}$$

whereas, with $\mathcal{X}_n := \{x_1, \dots, x_{n^*}\}$, the latter reads

$$(4.2) \quad \begin{aligned} \mathbb{P}^*(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k) : \text{maximize} \quad & \left\langle b, \sum_{i=1}^{k^*} (\gamma_i - \nu_i) y'_i \right\rangle + \varepsilon_k \sum_{i=1}^{k^*} (\gamma_i + \nu_i) \\ \text{subject to} \quad & \left\langle x_j, c - A' \sum_{i=1}^{k^*} (\gamma_i - \nu_i) y'_i \right\rangle \geq 0, \quad j = 1, \dots, n^*, \\ & \gamma_i, \nu_i \geq 0. \end{aligned}$$

Comparing (2.2) and (4.1) one may see that the vectors $y' \in \mathcal{Y}'$ in $\mathbb{P}^*$ are replaced by “inner” approximating vectors of the form

$$y' := \sum_{i=1}^{k^*} \alpha_i y'_i \in \text{sp}(\mathcal{Y}'_k) \subset \mathcal{Y}' \quad (\alpha_i \in \mathbb{R})$$

in $\mathbb{P}^*(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$. That is, to the aggregation-relaxation (3.1) of the primal problem $\mathbb{P}$ corresponds the inner approximation (4.1) of the dual $\mathbb{P}^*$. Moreover, the relaxation of the constraints in $\mathbb{P}$ to obtain $\mathbb{P}(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$ yields a *penalization* term

$$(4.3) \quad \varepsilon_k \sum_{i=1}^{k^*} (\gamma_i + \nu_i)$$

in the value of $\mathbb{P}^*(\mathcal{X}, \mathcal{Y}'_k, \varepsilon_k)$, that is, a penalization on the *magnitude* of the coefficients γ_i and ν_i (since, at an optimal solution, at most one of these coefficients is nonzero).

Similarly, if we “multiply” by $x_j \in \mathcal{X}_n$ the constraint vector

$$c - A' \sum_{i=1}^{k^*} (\gamma_i - \nu_i) y'_i \in \mathcal{X}'_+$$

in (4.1), we obtain the constraint (4.2). In other words, the constraints in (4.2) can be seen as “aggregated” forms of the constraints in (4.1), and, furthermore, the value of $\mathbb{P}^*(\mathcal{X}_n, \mathcal{Y}'_k, \varepsilon_k)$ also includes the penalization term (4.3).

Summarizing, to an *aggregation* of constraints in the primal problem corresponds an *inner approximation* of variables in the dual, and vice versa. Further, a *relaxation*

of the equality constraints in the primal yields a *penalization* of the corresponding variables in the value of the dual problem.

Example 4.1. The mass-transfer problem. Let X be a compact metric space, and let $M(X)$ and $\mathcal{X} := M(X \times X)$ be as in Remark 4.1. We define $P_1, P_2 : \mathcal{X} \rightarrow M(X)$ as the projection mappings, i.e., $(P_1\mu)(B) := \mu(B \times X)$ and $(P_2\mu)(B) := \mu(X \times B)$ for every measure μ in $\mathcal{X}$ and B in $\mathcal{B}(X)$. Let c be a given strictly positive function in $C(X \times X)$, and let $\nu_1, \nu_2 \in M_1(X)$ be two given probability measures on X . Then the *mass-transfer problem* is the linear program (cf. [1, 6, 14, 15, 21])

$$\begin{aligned} \text{MT : minimize } \langle \mu, c \rangle &:= \int_X \int_X c(x, y) \mu(d(x, y)) \\ \text{subject to } P_i \mu &= \nu_i \text{ for } i = 1, 2; \quad \mu \in \mathcal{X}_+. \end{aligned}$$

To put MT in the context of $\mathbb{P}$ and $\mathbb{P}'$ (see (2.1), (2.2)), consider the dual pairs $(\mathcal{X}, \mathcal{X}')$ and $(\mathcal{Y}, \mathcal{Y}')$ with $\mathcal{X} := M(X \times X)$ as above and

$$\mathcal{X}' := C(X \times X), \quad \mathcal{Y} := M(X) \times M(X), \quad \text{and} \quad \mathcal{Y}' := C(X) \times C(X).$$

In addition, let $A : \mathcal{X} \rightarrow \mathcal{Y}$ be the linear map defined as $A\mu := (P_1\mu, P_2\mu)$, whose adjoint $A' : \mathcal{Y}' \rightarrow \mathcal{X}'$ is

$$A'(f, g) := P'_1 f + P'_2 g,$$

where $P'_i : C(X) \rightarrow \mathcal{X}'$ is the adjoint of P_i ($i = 1, 2$), namely, $(P'_1 f)(x, y) := f(x)$ and $(P'_2 g)(x, y) := g(y)$. Note that A is *weakly continuous* since A' indeed maps $\mathcal{Y}'$ into $\mathcal{X}'$ (see the paragraph following Assumption 2.1). Then we can write MT and its dual as in (2.1) and (2.2):

$$\begin{aligned} \text{MT : minimize } \langle \mu, c \rangle \\ \text{subject to } A\mu &= (\nu_1, \nu_2), \quad \mu \in \mathcal{X}_+; \\ \text{MT}^* : \text{maximize } \langle (\nu_1, \nu_2), (f, g) \rangle &= \langle \nu_1, f \rangle + \langle \nu_2, g \rangle \\ \text{subject to } f(x) + g(y) &\leq c(x, y) \quad \forall x, y \in X, \quad f, g \in C(X). \end{aligned}$$

Observe that both programs are consistent (in particular, the product probability measure $\mu(B_1 \times B_2) := \nu_1(B_1)\nu_2(B_2)$ is feasible for MT) and that a feasible solution for MT is necessarily a *probability measure*. Moreover, the fact that X is a compact metric space implies that $\mathcal{X}$ is the *topological dual* of $\mathcal{X}'$ and, therefore (by the Alaoglu Theorem—see Remark 4.2), $M_1(X \times X)$ is sequentially compact in the weak* topology $\sigma(\mathcal{X}, \mathcal{X}')$; hence Assumption 2.1 holds. In fact, the same reason (X compact metric) yields that Assumptions 3.1 and 3.2 also hold since it implies that $\mathcal{Y}' := C(X) \times C(X)$ is separable and that, in addition, if $D \subset X \times X$ is a countable set dense in $X \times X$, then the set of measures whose supports are *finite* subsets of D is weakly dense in $M_1(X \times X)$ (see [2, p. 237, Thm. 4]). Summarizing, all of our assumptions are satisfied and, therefore, Theorems 2.1, 3.1, and 3.2 are applicable. Moreover, as Theorem 2.1(b) yields that there is no duality gap, Theorem 3.3 is also applicable.

Example 4.2. Markov control processes. A large class of optimal control problems (deterministic or stochastic, in discrete or continuous time, finite or infinite dimensional) can be posed as LP problems; see, for instance, [12, 14, 18, 19, 21] and references therein. Here we briefly describe a class of discrete-time stochastic control problems known as *Markov control processes* (MCPs); for full details, see [9, 10, 12].

An MCP can be represented as (X, A, Q, c) , where X is the *state space* and A is the *action (or control) set*, both assumed to be locally compact separable metric spaces; hence, so is their product $X \times A$. Further, the MCP's *transition law* Q , or $Q(B|x, a)$ with $B \in \mathcal{B}(X)$ and $(x, a) \in X \times A$, is a stochastic kernel on X given $X \times A$; that is, $Q(\cdot|x, a)$ is a probability measure on $\mathcal{B}(X)$ for every fixed (x, a) in $X \times A$, and $Q(B|\cdot)$ is a measurable function on $X \times A$ for every fixed B in $\mathcal{B}(X)$. Finally, $c : X \times A \rightarrow \mathbb{R}$ denotes the *one-step cost function*, and we will assume that

$$(4.4) \quad c \text{ is nonnegative and } \textit{inf-compact},$$

where the latter means that the level set $\{(x, a) \in X \times A | c(x, a) \leq r\}$ is compact for every $r \geq 0$. This implies that, in particular, c is *lower semicontinuous* (l.s.c.).

Now consider the dual pairs $(\mathcal{X}, \mathcal{X}')$ and $(\mathcal{Y}, \mathcal{Y}')$, where (using the notation in Remark 4.1 and with $|\mu|$ the *total variation* of a measure μ)

$$\mathcal{X} := \left\{ \mu \in M(X \times A) \mid \int cd|\mu| < \infty \right\},$$

$\mathcal{X}' := C_b(X \times A)$, $\mathcal{Y} := M(X) \times \mathbb{R}$, and $\mathcal{Y}' := C_0(X) \times \mathbb{R}$. If μ is in $\mathcal{X}$, we write its projection on X as $\mu_1(B) := \mu(B \times A)$ for $B \in \mathcal{B}(X)$ (cf. $P_1\mu$ in Example 4.1). Now let $L : \mathcal{X} \rightarrow M(X)$ be the linear map

$$(L\mu)(B) := \mu_1(B) - \int_{X \times A} Q(B|x, a)\mu(d(x, a)), \quad \mu \in \mathcal{X}, \quad B \in \mathcal{B}(X),$$

and define the linear map $T : \mathcal{X} \rightarrow \mathcal{Y}$ as $T\mu := (L\mu, \langle \mu, 1 \rangle)$. Then the so-called long-run expected *average cost* (AC) problem essentially reduces to the linear program

$$\text{AC: minimize } \langle \mu, c \rangle$$

$$(4.5) \quad \text{subject to } T\mu = (0, 1), \quad \mu \in \mathcal{X}_+.$$

Strictly speaking, AC is not of the same type as $\mathbb{P}$ since the cost function c is *not* necessarily in $\mathcal{X}'$; namely, under (4.4), c might not be continuous nor bounded. Nevertheless, by the definition of $\mathcal{X}$, the minimization problem still makes sense. On the other hand, to ensure that T is weakly continuous (equivalently, to guarantee that the adjoint T' maps $\mathcal{Y}'$ into $\mathcal{X}'$) we shall assume that, for every function u in $C_b(X)$, the function

$$(x, a) \mapsto \int_X u(y)Q(dy|x, a) \text{ is in } \mathcal{X}'.$$

(This is a Feller-like property.) Then the adjoint T' of T , which is given by

$$T'(h, \rho)(x, a) := \rho + h(x) - \int h(y)Q(dy|x, a), \quad (h, \rho) \in \mathcal{Y}',$$

maps $\mathcal{Y}'$ into $\mathcal{X}'$, and the dual program is

$$\text{AC*}: \text{maximize } \rho [= \langle (0, 1), (h, \rho) \rangle]$$

$$(4.6) \quad \text{subject to } T'(h, \rho) \leq c, \quad (h, \rho) \in \mathcal{Y}'.$$

It is obvious that AC^* satisfies Assumption 2.1(a); take, for instance, $\rho = 0$ and $h(\cdot) = 0$, and recall that c is nonnegative. With respect to Assumption 2.1(b), we shall suppose that there is a measure μ^0 that satisfies (4.5). Then combining this fact with the inf-compactness of c (see (4.4)) we obtain that the set $\mathcal{X}^0$ in Assumption 2.1(b), namely,

$$\mathcal{X}^0 := \{\mu \in \mathcal{X}_+ | \langle \mu, c \rangle \leq \langle \mu^0, c \rangle (< \infty)\},$$

is weakly sequentially *precompact* (see [3, p. 109] or [8]); that is, for every sequence $\{\mu_n\}$ in $\mathcal{X}^0$ there is a subsequence $\{\mu_k\}$ and a measure μ in $\mathcal{X}_+$ (but *not* necessarily in $\mathcal{X}^0$) such that $\langle \mu_k, f \rangle \rightarrow \langle \mu, f \rangle \forall f \in \mathcal{X}' := C_b(X \times A)$. On the other hand, as (by (4.4)) c is lower semicontinuous and nonnegative, there is a sequence $\{f_i\}$ in $\mathcal{X}'_+$ such that $f_i \uparrow c$, which, combined with the weak convergence $\mu_k \rightarrow \mu$ and the monotone convergence theorem, yields

$$\liminf_{k \rightarrow \infty} \langle \mu_k, c \rangle \geq \langle \mu, c \rangle.$$

Hence, μ is in $\mathcal{X}^0$ so that in fact $\mathcal{X}^0$ is weakly sequentially *compact*. Summarizing, the AC problem satisfies Assumption 2.1. Similar arguments are used in [12] to obtain results analogous to Theorems 3.1 and 3.2 for the average cost and other MCP-related problems. (Incidentally, the feasible solutions (h, ρ) in (4.6) to the dual problem AC^* are “subsolutions” of the *dynamic programming* equation for the AC problem. This is a fundamental fact that relates the LP formulation and the dynamic programming approach to optimal control problems.) Finally, as already noted in [12], we should mention that *all* the previous LP-based approximation schemes for MCPs assume that X and A are *compact* metric spaces

Example 4.3. Semi-infinite programming. A linear program is called *semi-infinite* if it consists of finitely many variables while the set of constraints is infinite. Let x and c be vectors in $\mathbb{R}^d$, and let $x^T c := \sum_{i=1}^d x_i c_i$ be the usual inner product. Moreover, let $K \subset \mathbb{R}$ be a compact (uncountable) set, and let $a : K \rightarrow \mathbb{R}^d$ and $b : K \rightarrow \mathbb{R}$ be given continuous functions. Then, following [1, p. 73], we consider the *continuous SI program*

$$\begin{aligned} \text{SI: minimize } & x^T c \\ \text{subject to } & x^T a(s) \geq b(s), s \in K. \end{aligned}$$

(If K is a *countable* set, then SI is said to be a *countable SI program*.) Note that SI is an *inequality* constrained problem [1, p. 38], as opposed to the equality constrained program $\mathbb{P}$ —see (2.1). We can now either put SI in the context of the previous sections (using slack variables to transform SI into the equality-constrained form (2.1)) or analyze the dual program SI^* . Choosing the latter option and using the notation in Remark 4.1, we then look at (cf. [1, p. 74])

$$(4.7) \quad \begin{aligned} SI^*: \text{ maximize } & \langle \mu, b \rangle \\ \text{subject to } & \langle \mu, a \rangle = c, \mu \in M(K)_+, \end{aligned}$$

where $\langle \mu, a \rangle$ is the d -vector with components $\langle \mu, a_i \rangle$, $i = 1, \dots, d$. This problem is basically the same as $\mathbb{P}$ except that the “minimization” in $\mathbb{P}$ is replaced by “maximization” in SI^* . In particular, introducing the dual pairs

$$(\mathcal{X}, \mathcal{X}') := (M(K), C(K)) \quad \text{and} \quad (\mathcal{Y}, \mathcal{Y}') := (\mathbb{R}^d, \mathbb{R}^d)$$

and the linear map $A : M(K) \rightarrow \mathbb{R}^d$ given by $A\mu := \langle \mu, a \rangle$, we may rewrite SI* as

$$\begin{aligned} \text{SI*}: \quad & \text{maximize } \langle \mu, b \rangle \\ & \text{subject to } A\mu = c, \mu \in M(K)_+. \end{aligned}$$

By its definition, it is obvious that A is *weakly continuous*, which can also be seen from the fact that the adjoint $A'x := x^T a$, with $x \in \mathbb{R}^d$, maps $\mathbb{R}^d$ into $C(K)$. We shall now assume that SI and SI* have the same (finite) values; that is, $\inf \text{SI} = \sup \text{SI}^*$. This is known to be true under several sets of conditions—see, for instance, Theorems 4.4 and 4.5 in [1]. There are also conditions to ensure that SI* is solvable, which would give that Theorem 2.1(a) holds for SI* in lieu of $\mathbb{P}$. For example, let us suppose that b is *strictly positive*, i.e.,

$$b_{\min} := \min_{s \in K} b(s) \geq \varepsilon \quad \text{for some } \varepsilon > 0.$$

Then, since $\langle \mu, b \rangle \geq \varepsilon \mu(K)$ for all μ in $M(K)_+$, the Alaoglu theorem referred to in Remark 4.2 guarantees that, for every positive number m , the set

$$\{\mu \in M(K)_+ \mid \langle \mu, b \rangle \leq m\}$$

is sequentially compact in the topology $\sigma(M(K), C(K))$. Then the same arguments used in the proof of Theorem 2.1(a), replacing “minimizing sequence” by “*maximizing* sequence,” show that indeed SI* is *solvable*. Finally, it is clear that the spaces $\mathcal{Y}' := \mathbb{R}^d$ and $\mathcal{X} := M(K)$ satisfy Assumptions 3.1 and 3.2, respectively. (Concerning $\mathcal{X}_+ = M(K)_+$, see the last part of Example 4.1 and observe that $\mu(\cdot)/\mu(K)$ is a *probability measure* on K for any nonnull measure μ in $M(K)_+$.) Therefore, our approximation results—Theorems 3.1 and 3.2—are applicable to SI*. For discretization schemes for SI (not SI*), see [1, 7, 13] and their references.

To conclude, it is worth noting that our approach can be used to approximately solve the cone-constrained linear equation $Ax = b$, with $x \in \mathcal{X}_+$, in (2.1). The idea would be simply to introduce an “artificial” cost vector $c \in \mathcal{X}'$ and then use the approximation procedure to solve the “optimization problem” $\mathbb{P}$. In fact, the use of LP techniques to solve different types of linear equations goes back at least to the early 1960s—see, for instance, [4, 16, 21] and their references.

Acknowledgment. We wish to thank the referees for their detailed and helpful comments, which greatly improved the paper.

REFERENCES

- [1] E. J. ANDERSON AND P. NASH, *Linear Programming in Infinite-Dimensional Spaces*, Wiley, Chichester, UK, 1987.
- [2] P. BILLINGSLEY, *Convergence of Probability Measures*, Wiley, New York, 1968.
- [3] N. BOURBAKI, *Intégration*, Hermann, Paris, 1969, Chap. IX.
- [4] T.-Y. CHEUNG, *Recent developments in the numerical solution of partial differential equations by linear programming*, SIAM Rev., 20 (1978), pp. 139–167.
- [5] B. D. CRAVEN AND J. J. KOLIHA, *Generalizations of Farkas’ theorem*, SIAM J. Math. Anal., 8 (1977), pp. 983–997.
- [6] V. GIRARDIN, *Extreme measures with given moments or marginals*, Canad. J. Math., 47 (1995), pp. 946–958.
- [7] M. A. GOBERNA AND M. A. LÓPEZ, *Optimality theory for semi-infinite linear programming*, Numer. Funct. Anal. Optim., 16 (1995), pp. 669–700.
- [8] J. GONZÁLEZ-HERNÁNDEZ AND O. HERNÁNDEZ-LERMA, *Envelopes of sets of measures, tightness, and Markov control processes*, Appl. Math. Optim., to appear.

- [9] O. HERNÁNDEZ-LERMA AND J. B. LASSERRE, *Discrete-Time Markov Control Processes: Basic Optimality Criteria*, Springer-Verlag, New York, 1996.
- [10] O. HERNÁNDEZ-LERMA AND J. B. LASSERRE, *Average optimality in Markov control processes via discounted-cost problems and linear programming*, SIAM J. Control Optim., 34 (1996), pp. 295–310.
- [11] O. HERNÁNDEZ-LERMA AND J. B. LASSERRE, *Cone-constrained linear equations in Banach spaces*, J. Convex Anal., 4 (1997), pp. 149–164.
- [12] O. HERNÁNDEZ-LERMA AND J. B. LASSERRE, *Linear programming approximations for Markov control processes in metric spaces*, Acta Appl. Math., 51 (1998), pp. 123–139.
- [13] R. HETTICH AND K. O. KORTANEK, *Semi-infinite programming: Theory, methods, and applications*, SIAM Rev., 35 (1993), pp. 380–429.
- [14] W. K. KLEIN HANEVELD, *Duality in Stochastic Linear and Dynamic Programming*, Lecture Notes Econ. Math. Syst. 274, Springer-Verlag, Berlin, 1986.
- [15] H. C. LAI AND S. Y. WU, *Linear programming in measure spaces*, Optimization, 29 (1994), pp. 141–156.
- [16] P. RABINOWITZ, *Applications of linear programming to numerical analysis*, SIAM Rev., 10 (1968), pp. 121–159.
- [17] A. P. ROBERTSON AND W. ROBERTSON, *Topological Vector Spaces*, Cambridge University Press, Cambridge, 1964.
- [18] J. RUBIO, *Control and Optimization: The Linear Treatment of Nonlinear Problems*, Manchester University Press, Manchester, and John Wiley, New York and London, 1986.
- [19] J. E. RUBIO, *The global control of nonlinear diffusion equations*, SIAM J. Control Optim., 33 (1995), pp. 308–322.
- [20] H. H. SCHAEFER, *Banach Lattices and Positive Operators*, Springer-Verlag, Berlin, 1974.
- [21] A. M. VERSHIK, *Some remarks on the infinite-dimensional problems of linear programming*, Russian Math. Surveys, 29 (1970), pp. 117–124.
- [22] A. M. VERSHIK AND V. TEMEL'T, *Some questions concerning the approximation of the optimal value of infinite-dimensional problems in linear programming*, Siberian Math. J., 9 (1968), pp. 591–601.

NONSMOOTH SHAPE OPTIMIZATION APPLIED TO LINEAR ACOUSTICS*

ABDERRAHMANE HABBAL†

Abstract. The purpose of this paper is to study the optimal design problem of a nonsmooth shape, applied to linear acoustics in a bounded domain. Within the classical Helmholtz model, we consider the maximum of the acoustical pressure in a prescribed sensitive area as cost function to be minimized. A given part of the boundary of the domain is considered as the design variable. The existence of subgradients is demonstrated, and a technique for computing them is described. It turns out that the classical (differentiable) sensitivity subroutines could be used in order to compute subgradients.

The numerical results presented tend to show that the acoustical pressure is not very sensitive to small changes of the shape for low frequencies, but it is sensitive for medium frequencies, and one obtains an optimal shape for the L^2 cost that is slightly different from the one obtained for the L^∞ cost.

Key words. acoustics, shape optimization, subgradients

AMS subject classifications. 65K10, 76Q05, 49K20

PII. S105262349529581X

1. Introduction. The noise reduction problem is of great interest in several industries, such as the aerospace, automobile, and machine industries, and in freeway engineering.

There are, generally speaking, two different approaches:

- active noise reduction—based on sound cancellation techniques, using tools from various disciplines, mainly from microelectronics and digital signal processing (see, e.g., [1]).
- passive noise reduction—consisting of seeking the best damping material or the best thickness distribution for a multilayer medium (see, e.g., [2]) or, for a given material, looking for the optimal shape.

We consider the passive approach consisting of looking for the best shape of a given part of the geometry of the whole system (e.g., a car component) that minimizes the annoying noise effects (on human beings or on any sensitive area).

From a theoretical point of view, it is simpler to minimize differentiable costs such as the square of the L^2 -norm of the acoustical pressure (or any relevant acoustical quantity). However, one may be interested in minimizing costs that may be more interesting than the differentiable ones but are not smooth.

This situation is classical in structural mechanics when considering, e.g., aging fatigue or crack or fracture of mechanical structures, where the most interesting criterion is the maximum of mechanical stresses over the structure (which is not differentiable with respect to the shape of the structure). See [3] for a complete investigation of the problem for a distributed parameter elliptic system.

From the acoustics point of view, the amplitude of a sound is usually measured by means of the *sound pressure level* (SPL) [1], which is a *local* definition (given for each point of the space). Then critical concentration of SPL may cause damage just

*Received by the editors December 4, 1995; accepted for publication (in revised form) July 8, 1997; published electronically September 10, 1998.

<http://www.siam.org/journals/siopt/8-4/29581.html>

†Laboratoire Jean-Alexandre Dieudonné, UMR 6621 CNRS, Parc Valrose, University of Nice-Sophia Antipolis, 06108 Nice Cedex, France (habbal@unice.fr).

as mechanical stress does for structures. Minimizing the maximal SPL in the entire sensitive area may avoid such an undesired concentration.

Section 2 presents the acoustical model and defines the optimal design problem. Then, in section 3, the definition of derivative with respect to the domain variable is stated. As a nonsmooth cost function is considered in section 4, some basic notions from nonsmooth optimization theory (NSO) are recalled. Then, in section 5, the existence of subgradients for the considered cost is demonstrated, and a method using existing routines (for the differentiable case) in order to compute them is presented in section 6. Finally, numerical tests for the sensitivity analysis and several optimization results are presented in section 7.

2. The acoustical model and its optimal design approach.

2.1. The acoustics model. The propagation of acoustical waves of small amplitude in a compressible, inviscid, isotropic homogeneous fluid is governed by the linearized Euler equations; see, e.g., [4].

When the emitting sources are time-harmonic, the resulting acoustical pressure is, too. We assume, for simplicity, that the domain occupied by the fluid, denoted by Ω , is bounded with a *sound-hard* boundary. Then, if the acoustical source is of the form

$$S(x, t) = S(x) \cos(\omega t),$$

the pressure field solution of the linearized Euler equation is of the same form, given by

$$p(x, t) = p(x) \cos(\omega t),$$

where x denotes the spatial variable, t the time variable, and ω the pulsation of the source.

The unknown of the problem is now $p(x)$, which is the solution to the Helmholtz equation

$$(2.1) \quad \begin{cases} -\Delta p - k^2 p = S & \text{in } \Omega, \\ \frac{\partial p}{\partial n} = 0 & \text{over } \partial\Omega, \end{cases}$$

with $k = \frac{\omega}{c}$ being the wave number, and c the speed of sound inside the fluid.

Since the domain Ω is bounded, the linear operator $-\Delta$ has real, positive eigenvalues depending on Ω . We assume that k^2 is not one of the eigenvalues, which implies that for any source $S \in L^2(\Omega)$ there exists one and only one solution $p \in H^1(\Omega)$ to the Helmholtz equation (2.1); see, e.g., [5].

REMARK 2.1. *It is shown in [6] that the mapping $\Omega \mapsto \lambda_\Omega$, where λ_Ω is an eigenvalue of the Laplace operator, is continuous. Hence, if k^2 is not an eigenvalue for a given Ω , it is not one for any sufficiently small perturbation of the open set Ω .*

The optimization process will perturb the initial domain Ω , giving a new domain Ω' , and we have to check, numerically at least, that k^2 is not an eigenvalue of Ω' .

2.2. The optimal design approach. The optimal design problem is the following.

Let Ω be a bounded open subset of $\mathbb{R}^2$, with a regular enough boundary $\partial\Omega = \Gamma_1 \cup \Gamma_2$.

A time-harmonic acoustical source, located inside Ω , generates an acoustical pressure p_Ω , which depends on the shape of the domain Ω .

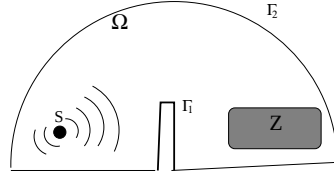


FIG. 2.1. Example of domain geometry.

Given a subset Z of Ω , the optimization problem is to find the shape of Γ_1 which minimizes a *cost function*, depending explicitly on p_Ω and possibly on Ω .

The problem posed above is an approximation of real-world situations, where the source of noise is machinery, an air duct, or any noise-generating system, and Z is a region to protect against the noise annoyance. See Figure 2.1.

The cost function is the maximum of the square of the acoustical pressure over the sensitive area Z given by

$$j_\infty(\Omega) = \max_{x \in \overline{Z}} |p_\Omega(x)|^2.$$

This cost function is well defined since p_Ω is continuous over $\overline{Z}$; see section 5.2.

In order to minimize such a function, classical optimization algorithms use some numerical approximation of the gradient (whenever it exists) as the descent direction.

As we consider a set of design variables (the set of domains of $\mathbb{R}^2$) that has no vector space structure, and a cost function which is a priori not “differentiable,” two basic notions have to be stated:

- the notion of *differentiability* with respect to the domain variable,
- the notion of *subdifferentiability* for nonsmooth functions.

3. The concept of domain derivative.

Let us introduce some notation.

- $\hat{\Omega}$ is a fixed open bounded subset of $\mathbb{R}^2$, with a piecewise C^1 boundary.
- $\hat{W} = W^{1,\infty}(\hat{\Omega}; \mathbb{R}^2)$ is equipped with its natural norm

$$\|\hat{T}\|_{\hat{W}} = \|\hat{T}\|_{L^\infty(\hat{\Omega})} + \|D\hat{T}\|_{L^\infty(\hat{\Omega})}.$$

- $\hat{U} = \{\hat{T} \in \hat{W}, \hat{T} \text{ is bijective, } \hat{T}^{-1} \in W^{1,\infty}(\hat{T}(\hat{\Omega}); \mathbb{R}^2)\}$, and $\hat{U}$ is an open subset of $\hat{W}$.

The *set of topologically admissible domains* is defined by

$$D(\hat{\Omega}) = \{\Omega = \hat{T}(\hat{\Omega}) / \hat{T} \in \hat{U}\}.$$

Let j be a generic domain functional such that

$$\begin{aligned} j &: D(\hat{\Omega}) \longrightarrow \mathbb{R}, \\ \Omega &\longmapsto j(\Omega). \end{aligned}$$

We then associate a functional $\hat{j}$ to j , defined by

$$\begin{aligned} (3.1) \quad \hat{j} &: \hat{U} \longrightarrow \mathbb{R}, \\ \hat{T} &\longmapsto \hat{j}(\hat{T}) = j(\Omega), \\ &\text{where } \Omega = \hat{T}(\hat{\Omega}). \end{aligned}$$

The functional $\hat{j}$ is now defined over an open subset of the Banach space $\hat{W}$, and we can define the *Frechet derivative* of $\hat{j}$, at any element $\hat{T}_0 \in \hat{U}$.

The *domain derivative* of j at a given $\Omega_0 \in D(\hat{\Omega})$ is therefore defined in the following.

DEFINITION 3.1. *Let $\hat{T}_0 \in \hat{U}$ such that $\hat{T}_0(\hat{\Omega}) = \Omega_0$. The functional j is said to be differentiable at Ω_0 if the mapping $\hat{j}$ is Frechet differentiable at $\hat{T}_0$. We then set*

$$(3.2) \quad \frac{d}{d\Omega} j(\Omega_0) \cdot V = \frac{d}{d\hat{T}} \hat{j}(\hat{T}_0) \cdot (V \circ \hat{T}_0) \quad \forall V \in W^{1,\infty}(\Omega_0; \mathbb{R}^2).$$

(The notation $a.b$ denotes the duality product, and $a \circ b$ is the composition of the mappings a and b .)

REMARK 3.1. *The domain derivative defined above does not depend on the mapping that represents the domain Ω_0 , in the sense that any other mapping $\hat{T}_1$ such that $\hat{T}_1(\hat{\Omega}) = \Omega_0$ leads to the same domain derivative.*

From now on, any notion of continuity or differentiability of a domain functional

$$f : \Omega \longrightarrow f(\Omega) : D(\hat{\Omega}) \longrightarrow \mathbb{R}$$

is to be understood in the sense of Definition 3.1, using the corresponding *hat* mapping

$$\hat{T} \longrightarrow \hat{f}(\hat{T}) = f(\hat{T}(\hat{\Omega})) : \hat{U} \longrightarrow \mathbb{R}.$$

4. Basic tools from NSO. NSO deals with cost functions which are not necessarily differentiable (in the Gateaux or Frechet sense).

Within the framework of infinite-dimensional Banach spaces, the notion of the subdifferential is briefly defined. Then, a theorem related to the particular case of maximum-like functions is presented, and a short description of the NSO algorithm principle is given. For more details, readers may refer to [7], [8], [9], [10], and [11].

4.1. Subdifferentials. Let W be a Banach space, and let f be a mapping $f : W \longrightarrow \mathbb{R}$.

The *generalized directional derivative in the sense of Clarke* of f at $x \in W$, in the direction $d \in W$, is given by

$$f^\circ(x; d) = \limsup_{\substack{y \rightarrow x \\ t \rightarrow 0^+}} \frac{f(y + td) - f(y)}{t}.$$

Assume that f is Lipschitzian at $x \in W$. Then, it is known that $f^\circ(x; \cdot)$ exists and is a positively homogeneous mapping, subadditive over W . Hence, from the Hahn–Banach theorem, this mapping is an upper bound for some linear form $g \in W'$, the dual space of W .

The *subdifferential in the sense of Clarke*, $\partial f(x)$, of f at x is then defined as

$$\partial f(x) = \{g \in W' \text{ s.t. } \forall d \in W, f^\circ(x; d) \geq \langle g, d \rangle_{W', W}\}.$$

The elements of $\partial f(x)$ are called *subgradients* of f at x .

When f is strictly (or continuously) differentiable, $\partial f(x)$ reduces to the singleton $\{df/dx(x)\}$ [10].

4.2. Maximum functions. A maximum function is of the type

$$f(x) = \max \{h(x, y) ; y \in Y\}.$$

Let B be an open set of a Banach space W , T a separable topological space, and Y a compact subset of T . For any subset of T , the collection of probability Radon measures carried by S is denoted by $P[S]$.

Let $h : B \times Y \rightarrow \mathbb{R}$ be a function such that

1. $(x, y) \rightarrow h(x, y)$ is a continuous mapping over $B \times Y$,
2. $x \rightarrow h(x, y)$ is differentiable over B , for any $y \in Y$,
3. $(x, y) \rightarrow \frac{\partial h}{\partial x}(x, y)$ is continuous over $B \times Y$.

Let us set

$$X(x) = \{y \in Y; f(x) = h(x, y)\},$$

where f is the maximum function defined above.

Then, the subdifferential $\partial f(x)$ of f exists for any x [10] and is given by

$$\partial f(x) = \left\{ \int_Y \frac{\partial h}{\partial x}(x, z) d\mu_z; \mu \in P[X(x)] \right\}.$$

REMARK 4.1. *The space $P[X(x)]$ contains in particular all the Dirac measures defined over $X(x)$. By setting $\mu = \delta_z$, one deduces that for any $z \in X(x)$, the differential $\frac{\partial h}{\partial x}(x, z)$ is one of the subgradients of f at x .*

4.3. The algorithms. NSO deals with functions whose gradients are not necessarily continuous. In this case, classical methods fail. New tools are then required to replace standard differentiability-based methods. One of the most recent and sophisticated NSO method-classes is the so-called bundle method [12] in which the descent direction is computed by projecting the origin onto a convex polyhedron generated by a set of subgradients.

In general, NSO subroutines require the user to supply, for a given input variable, the cost value and only one subgradient.

5. Application to the Helmholtz model. It is assumed that the reader is familiar with basic theory of optimal control, more specifically the *adjoint state method* used to compute the gradient [13], [14], [15].

Within the optimal design framework set in the previous section, we consider a given reference domain $\hat{\Omega}$, e.g., the one of Figure 2.1 with a rectangular barrier Γ_1 .

The space of controls $\hat{W}$, the open subset $\hat{U}$ of $\hat{W}$, and the set $D(\hat{\Omega})$ of admissible domains are given by the following.

- $\hat{W} = W^{1,\infty}(\hat{\Omega}; \mathbb{R}^2)$,
- $\hat{U} = \{\hat{T} \in \hat{W}, \hat{T} \text{ is bijective, } \hat{T}^{-1} \in W^{1,\infty}(\hat{T}(\hat{\Omega}); \mathbb{R}^2)\}$,
- $D(\hat{\Omega}) = \{\Omega = \hat{T}(\hat{\Omega}) / \hat{T} \in \hat{U}\}$.

5.1. The state equation. To any domain $\Omega \in D(\hat{\Omega})$, we associate a *direct state* equation

$$(5.1) \quad \begin{cases} -\Delta p_\Omega - k^2 p_\Omega = S_\Omega & \text{in } \Omega, \\ \frac{\partial p_\Omega}{\partial n} = 0 & \text{over } \partial\Omega. \end{cases}$$

If the acoustical source S_Ω belongs to the space $L^2(\Omega)$, then the solution p_Ω belongs to the space $\mathcal{H}(\Omega)$ defined by

$$\mathcal{H}(\Omega) = \left\{ v \in H^1(\Omega) \cap H^2(\Omega'); \frac{\partial v}{\partial n} = 0 \right\},$$

where $\Omega' \subset \overline{\Omega'} \subset \Omega$ is any open regular set, sufficiently far from the boundary of Ω ; see, e.g., [16, p. 197]. From now on, we choose the subset Ω' such that $\overline{Z} \subset \Omega'$.

The strong linear operator $A(\Omega)$ associated with the Helmholtz equation

$$(5.2) \quad A(\Omega) : \mathcal{H}(\Omega) \longrightarrow L^2(\Omega),$$

$$(5.3) \quad p \longmapsto A(\Omega).p = -\Delta p - k^2 p$$

is then an *isomorphism* from $\mathcal{H}(\Omega)$ onto $L^2(\Omega)$.

Since the mapping $\Omega \longmapsto A(\Omega)$ is differentiable [17], we conclude using the implicit function theorem that, provided the mapping $\Omega \longmapsto S_\Omega$ is differentiable as well, the mapping $\Omega \longmapsto p_\Omega$ is differentiable from $D(\hat{\Omega})$ onto $\mathcal{H}(\Omega)$ and hence, more strongly, from $D(\hat{\Omega})$ onto the Sobolev space $H^1(\Omega)$. The equivalent weak form

$$(5.4) \quad \exists! p_\Omega \in H^1(\Omega), \quad \forall q \in H^1(\Omega), \quad a(\Omega; p_\Omega, q) = L(\Omega; q),$$

where

$$\begin{aligned} a(\Omega; p, q) &= \int_\Omega \nabla p \nabla q d\Omega - k^2 \int_\Omega p \cdot q d\Omega, \\ L(\Omega; q) &= \int_\Omega S \cdot q d\Omega, \end{aligned}$$

is usually used when the observation is differentiable, in order to derive the adjoint state equation and to compute the gradient of the cost function.

5.2. Subdifferentiability of the cost function. The cost function is given by

$$j_\infty(\Omega) = \max_{x \in \overline{Z}} |p_\Omega(x)|^2,$$

where p_Ω is the solution to the Helmholtz equation (5.1).

This cost is well defined since the state variable p_Ω belongs to the space $\mathcal{H}(\Omega)$ (hence, from the Sobolev embedding theorem, p_Ω is continuous over the subset $\overline{Z} \subset \Omega'$).

In a first step, the hat correspondence is used in order to rewrite j_∞ as a maximum of a two-argument function. Then, we prove that, for our problem, the three hypotheses of section 4.2 hold, which implies the existence of subgradients of j_∞ at any admissible domain.

Since the *target element*, i.e., the geometrical piece of the domain to redesign, is the barrier Γ_1 and *not* the sensitive area Z , we assume that this subset is stable for any mapping $\hat{T} \in \hat{U}$, i.e., $\hat{T}(\hat{Z}) = \hat{Z} = Z$. Thus, j_∞ could be rewritten as

$$\begin{aligned} j_\infty(\Omega) &= \max \{ |p_\Omega(z)|^2; z \in \overline{Z} \} \\ &= \max \{ |p_\Omega(\hat{z})|^2; \hat{z} \in \overline{\hat{Z}} \} \\ &= \max \{ |\hat{p}(\hat{T}; \hat{z})|^2; \hat{z} \in \overline{\hat{Z}} \} \\ &= \hat{j}_\infty(\hat{T}), \end{aligned}$$

where, by definition, $\hat{p}(\hat{T}; \hat{z}) = p_\Omega(\hat{z})$.

Now, one could observe that, from section 4.2, the regularity properties of the mapping $(\hat{T}; \hat{z}) \longrightarrow \hat{p}(\hat{T}; \hat{z})$ play a central role in the existence of subgradients of j_∞ .

PROPOSITION 5.1. *The following properties are fulfilled.*

- H1 $(\hat{T}; \hat{z}) \longrightarrow \hat{p}(\hat{T}; \hat{z})$ is a continuous mapping over $\hat{U} \times \overline{\hat{Z}}$;

- H2 $\hat{T} \longrightarrow \hat{p}(\hat{T}; \hat{z})$ is differentiable over $\hat{U}$, for any $\hat{z} \in \overline{\hat{Z}}$;
- H3 $(\hat{T}; \hat{z}) \longrightarrow \frac{\partial \hat{p}}{\partial \hat{T}}(\hat{T}; \hat{z})$ is continuous over $\hat{U} \times \overline{\hat{Z}}$.

Then, by applying the results of section 4.2 and Remark 4.1, one has the following corollary.

COROLLARY 5.2. *The cost function j_∞ has subgradients at any $\Omega \in D(\hat{\Omega})$. Moreover, for any $\hat{x} \in X(\Omega)$, where*

$$X(\Omega) = \{\hat{x} \in \hat{Z} \text{ s.t. } j_\infty(\Omega) = |p_\Omega(\hat{x})|^2\},$$

the quantity $g(\Omega)$ defined by

$$(5.5) \quad g(\Omega).V = \frac{\partial |p_\Omega(\hat{x})|^2}{\partial \Omega}.V = 2.p_\Omega(\hat{x}).\left(\frac{\partial p_\Omega}{\partial \Omega}.V\right)(\hat{x}), \quad V \in W^{1,\infty}(\Omega; \mathbb{R}^2),$$

is a subgradient of j_∞ at Ω .

Proof of Proposition 5.1. If necessary, the correspondence between the mappings

$$(\hat{T}; \hat{z}) \longrightarrow \hat{p}(\hat{T}; \hat{z})$$

and

$$(\Omega; \hat{z}) \longrightarrow p_\Omega(\hat{z})$$

will be implicitly used.

Proof of H2. The mapping $\Omega \longrightarrow p_\Omega(\hat{z})$ is differentiable, since it is the composition of the mappings

$$\Omega \longrightarrow p_\Omega \in \mathcal{H}(\Omega),$$

which is continuously differentiable (from the implicit function theorem), and

$$q \longrightarrow q(\hat{z}),$$

which is a linear continuous form over $\mathcal{H}(\Omega)$, since $\hat{z} \in \overline{\hat{Z}} \subset \Omega'$.

Proof of H1. Let $(\hat{T}_0; \hat{z}_0) \in \hat{U} \times \overline{\hat{Z}}$, $\hat{T}_0(\hat{\Omega}) = \Omega_0$.

• Since, from the previous proof of H2, the partial mapping $\hat{T} \longrightarrow \hat{p}(\hat{T}; \cdot)$ is continuous, and $C^0(\overline{\hat{Z}})$ is continuously embedded in $\mathcal{H}(\Omega)$, one has

$$(5.6) \quad \forall \epsilon > 0, \exists \delta_1(\epsilon, \hat{T}_0) > 0 \text{ s.t.} \\ ||\hat{T} - \hat{T}_0||_{\hat{W}} < \delta_1 \Rightarrow \forall \hat{z} \in \overline{\hat{Z}}, |\hat{p}(\hat{T}; \hat{z}) - \hat{p}(\hat{T}_0; \hat{z})| < \epsilon/2.$$

• Since the mapping $\hat{p}(\hat{T}_0; \cdot)$ is continuous over the compact set $\overline{\hat{Z}}$, it is uniformly continuous, so that

$$(5.7) \quad \forall \epsilon > 0, \exists \delta_2(\epsilon, \hat{T}_0) > 0 \text{ s.t.} \\ \forall \hat{z}, \hat{z}_0 \in \overline{\hat{Z}}, |\hat{z} - \hat{z}_0| < \delta_2 \Rightarrow |\hat{p}(\hat{T}_0; \hat{z}) - \hat{p}(\hat{T}_0; \hat{z}_0)| < \epsilon/2.$$

The continuity of the mapping $(\hat{T}; \hat{z}) \longrightarrow \hat{p}(\hat{T}; \hat{z})$, uniform in $\hat{z}$, is then straightforward.

Proof of H3. The technique is similar to the one used to prove H1.

Let $(\hat{T}_0; \hat{z}_0) \in \hat{U} \times \overline{\hat{Z}}$, $\hat{T}_0(\hat{\Omega}) = \Omega_0$.

- The partial differential mapping

$$\hat{T}_0 \longrightarrow \frac{\partial \hat{p}}{\partial \hat{T}}(\hat{T}_0; \cdot) \in \mathcal{L}(W^{1,\infty}(\Omega_0; \mathbb{R}^2); \mathcal{H}(\Omega))$$

is continuous, since $\Omega \longrightarrow p_\Omega$ is of class C^1 (the isomorphism $A(\Omega)$ is of class C^1 with respect to the domain, in the sense of Definition 3.1; we assume that the source is, too).

- The partial differential mapping

$$\hat{z} \longrightarrow \frac{\partial \hat{p}}{\partial \hat{T}}(\hat{T}_0; \hat{z})$$

is continuous (hence uniformly continuous) over $\overline{\hat{Z}}$.

Let $\hat{z}_n \rightarrow \hat{z}$ be a sequence in $\overline{\hat{Z}}$. One has

$$\forall V \in W^{1,\infty}(\Omega_0; \mathbb{R}^2), \quad \left(\frac{\partial \hat{p}}{\partial \hat{T}} \cdot V \right) (\hat{T}_0; \cdot) \in \mathcal{H}(\Omega_0),$$

which is then continuous over $\overline{\hat{Z}}$, so that

$$\left(\frac{\partial \hat{p}}{\partial \hat{T}} \cdot V \right) (\hat{T}_0; \hat{z}_n) \rightarrow \frac{\partial \hat{p}}{\partial \hat{T}} \cdot V (\hat{T}_0; \hat{z}) \text{ in } \mathbb{R}$$

i.e.,

$$\frac{\partial \hat{p}}{\partial \hat{T}}(\hat{T}_0; \hat{z}_n) \rightarrow \frac{\partial \hat{p}}{\partial \hat{T}}(\hat{T}_0; \hat{z}) \text{ in } W^{1,\infty}(\Omega_0; \mathbb{R}^2)'.$$

Then the same technique previously used applies here.

5.3. An adjoint state method in a very weak sense. From the previous section, it is obvious that a particular subgradient of j_∞ at a given domain Ω is exactly the gradient of the pointwise function

$$j_0(\Omega) = |p_\Omega(z)|^2$$

at a *fixed* point $z \in Z$.

Recall that from hypothesis H2 the function j_0 is differentiable at any admissible Ω . One could then use an adjoint state method in order to compute the domain derivative of j_0 , i.e., a subgradient of j_∞ .

If we denote by $J_0(p) = |p(z)|^2$ the associated observation to j_0 , i.e., $j_0(\Omega) = J_0(p_\Omega)$, then the classical adjoint state equation has, as right-hand side, the mapping

$$w \longrightarrow \frac{\partial J_0}{\partial p}(p_\Omega) \cdot w = 2 \cdot p_\Omega(z) \cdot w(z),$$

which is a linear continuous form over the space $\mathcal{H}(\Omega)$. But it is *not* an element of $L^2(\Omega)$.

To overcome this difficulty, we define an adequate adjoint state equation, then use the transposition method [16] to prove the existence of a unique solution.

The strong form of the Helmholtz equation

$$(5.8) \quad \forall S_\Omega \in L^2(\Omega), \exists! p_\Omega \in \mathcal{H}(\Omega) \text{ s.t. } A(\Omega).p_\Omega = S_\Omega$$

is equivalent to the form

$$(5.9) \quad \begin{aligned} & \forall S_\Omega \in L^2(\Omega), \exists! p_\Omega \in \mathcal{H}(\Omega) \text{ s.t.} \\ & \forall q \in L^2(\Omega) \langle A(\Omega).p_\Omega, q \rangle_{L^2(\Omega)} = \langle S_\Omega, q \rangle_{L^2(\Omega)}. \end{aligned}$$

Since the linear operator $A(\Omega) : \mathcal{H}(\Omega) \rightarrow L^2(\Omega)$ is an isomorphism, its adjoint operator $A(\Omega)^* : L^2(\Omega) \rightarrow \mathcal{H}(\Omega)'$ is, too. Then one has the following proposition.

PROPOSITION 5.3. *There exists one and only one solution $p_\Omega^* \in L^2(\Omega)$ of the equation*

$$(5.10) \quad \forall w \in \mathcal{H}(\Omega) \quad \langle w, A(\Omega)^*.p_\Omega^* \rangle_{\mathcal{H}(\Omega), \mathcal{H}(\Omega)'} = \frac{\partial J_0}{\partial p}(p_\Omega).w.$$

The solution p_Ω^* , called the adjoint state in a very weak sense, belongs to the space $L^2(\Omega) \cap H^1(K)$, where K is any arbitrary open subset of Ω , such that $K \cap Z = \emptyset$.

The existence and uniqueness of the solution p_Ω^* are straightforward. The local regularity property is due to the fact that pointwise sources appearing in the adjoint state equation (5.10) are all localized in the sensitive area Z . Then, from classical elliptic problem theory (see, e.g., [16]), one has H^1 regularity far from the sources.

Now using the classical adjoint state method applied to the *formal Lagrangian*

$$L(\Omega; p, q) = J_0(p) - \langle A(\Omega).p, q \rangle_{L^2(\Omega)} + \langle S_\Omega, q \rangle_{L^2(\Omega)},$$

one obtains

$$(5.11) \quad \begin{aligned} \frac{\partial j_0}{\partial \Omega}(\Omega).V &= \frac{\partial L}{\partial \Omega}(\Omega; p_\Omega, p_\Omega^*).V \\ &= -\frac{\partial}{\partial \Omega} \langle A(\Omega).p_\Omega, p_\Omega^* \rangle_{L^2(\Omega)}.V + \frac{\partial}{\partial \Omega} \langle S_\Omega, p_\Omega^* \rangle_{L^2(\Omega)}.V \end{aligned}$$

for any $V \in W^{1,\infty}(\Omega; \mathbb{R}^2)$.

Note that the mathematical framework stated in this section is independent of the nonsmoothness aspect of the cost function. It may be useful whenever the observation being considered requires more regularity of the direct state variable than usually given by the existence theorems for the state equation.

5.4. Practical computation of a subgradient. The algorithm used in order to compute a subgradient of the nonsmooth cost function j_∞ is the following:

1. Compute p_Ω , the solution of the Helmholtz equation (5.8).
2. Compute $\max_{z \in \bar{Z}} |p_\Omega(z)|^2$; select one z_0 where the maximum is attained.
3. Compute p_Ω^* , the solution of the adjoint state equation (5.10), where the right-hand side is $2.p_\Omega(z_0).w(z_0)$.
4. Then a subgradient $g(\Omega)$ of j_∞ at Ω is given by formula (5.11), which can be written more explicitly [13], [18] as

$$(5.12) \quad \begin{aligned} g(\Omega).V &= -\frac{\partial}{\partial \Omega} \left(\int_\Omega -\Delta p_\Omega.p_\Omega^* d\Omega \right).V \\ &\quad - k^2 \int_\Omega p_\Omega.p_\Omega^*.\text{div}(V) d\Omega + \int_\Omega S_\Omega.p_\Omega^*.\text{div}(V) d\Omega, \end{aligned}$$

for any $V \in W^{1,\infty}(\Omega; \mathbb{R}^2)$, s.t. $V \equiv 0$ over $\bar{Z}$.

In the above formula (5.12), the first term in the right-hand side has not been made explicit, because it is not a standard one. In the classical situations involving the Laplace operator, the adjoint state p_Ω^* is usually of H^1 regularity, so that Green's formula and boundary conditions lead to the symmetric term

$$\int_{\Omega} -\Delta p_{\Omega} \cdot p_{\Omega}^* d\Omega = \int_{\Omega} \nabla p_{\Omega} \cdot \nabla p_{\Omega}^* d\Omega,$$

whose partial derivative is given by

$$\frac{\partial}{\partial \Omega} \left(\int_{\Omega} \nabla p_{\Omega} \cdot \nabla p_{\Omega}^* d\Omega \right) \cdot V = \int_{\Omega} \nabla p_{\Omega} \cdot \nabla p_{\Omega}^* \operatorname{div}(V) d\Omega - \int_{\Omega} \langle [DV] \nabla p_{\Omega}, \nabla p_{\Omega}^* \rangle d\Omega, \quad (5.13)$$

where $[DV] = (DV) + (DV)^T$ and $(DV)_{ij} = dV_i/dx_j$.

The heavy formula (5.13) is currently the most frequently implemented formula in shape sensitivity analysis subroutines. It would be then of interest to have the ability to use the same subroutine in order to compute subgradients of j_∞ .

In the present case this is possible, since one has $V \equiv 0$ in $\bar{Z}$ (the shape of the sensitive area Z does not change), and all the integrals over Ω in formula (5.12) reduce to integrals over Ω/Z . Since, from Proposition 5.3, the adjoint state in a very weak sense p_Ω^* belongs to the space $H^1(\Omega/Z)$, Green's formula applies.

Finally, the expression of a subgradient $g(\Omega)$ of j_∞ at Ω is given by

$$\begin{aligned} \forall V \in W^{1,\infty}(\Omega; \mathbb{R}^2), \text{ s.t. } V \equiv 0 \text{ in } \bar{Z} \\ g(\Omega) \cdot V = \int_{\Omega/Z} \nabla p_{\Omega} \cdot \nabla p_{\Omega}^* \operatorname{div}(V) d\Omega - \int_{\Omega/Z} \langle [DV] \nabla p_{\Omega}, \nabla p_{\Omega}^* \rangle d\Omega \\ - k^2 \int_{\Omega/Z} p_{\Omega} \cdot p_{\Omega}^* \cdot \operatorname{div}(V) d\Omega + \int_{\Omega/Z} S_{\Omega} \cdot p_{\Omega}^* \cdot \operatorname{div}(V) d\Omega. \end{aligned} \quad (5.14)$$

6. Numerical approximations. Three different types of approximation are necessary when implementing software within the shape optimal design framework:

- the *domain approximation*, which consists of discretizing the infinite-dimensional space of controls, using geometrical approximation tools, e.g., *cubic splines*, to approximate the boundary of the domain;
- the *direct* and *adjoint state approximation*;
- the *gradient* or *subgradient approximation*.

Since, in our case, subgradients appear as gradients of smooth functions, we will not distinguish between them, talking about “gradient approximation” in both cases. The main difference concerns the *minimization algorithm* used to find an approximate optimum, which must be chosen from the NSO toolbox in the nonsmooth case; see section 4.

6.1. The domain approximation. In a first step, we choose to approximate the general shape of the target element by *cubic splines*, as shown in Figure 6.1, using a finite number N of interpolation nodes, which are called *master nodes*.

The optimization variables now become the master nodes coordinates.

In order to build a finite-dimensional space W_N as an approximation to the space $W^{1,\infty}(\Omega/Z; \mathbb{R}^2)$, we first compute the N basic elements V_i , $i = 1, \dots, N$, such that

$$(6.1) \quad \begin{cases} -\Delta V_i = 0 & \text{over } \Omega/Z, \\ V_i = 0 & \text{over } \Gamma_2 \cup \partial Z, \\ V_i = s_i & \text{over } \Gamma_1, \end{cases}$$

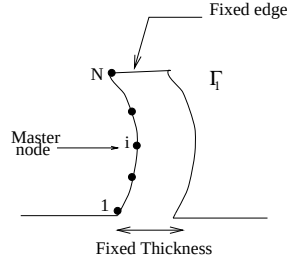


FIG. 6.1. Approximation of the target element.

where the functions s_i are N basic cubic splines depending on the N master nodes.

The discrete space W_N is then the space spanned by the V_i 's, i.e., the successive domains will be of the form

$$\Omega^{k+1} = \Omega^k + \sum_{i=1}^{i=N} \alpha_i V_i(\Omega^k).$$

6.2. The state variables approximation. The method used to compute approximations of the V_i defined above, the acoustical pressure, and its adjoint state is the finite element method (FEM).

In this paper, we used piecewise-linear triangular finite element approximation, which is of first-order precision. The choice of this low-order method is guided by the fact that the optimization process needs many FEM computations, and one has to make a compromise between cheap computations and precise results.

6.3. The gradient approximation. The exact derivative of j_0 with respect to the domain variable is given by the formula (5.11). This derivative is called a continuous derivative, in the sense that it is directly derived from the continuous model. We define the discretized continuous gradient (DCG) as a numerical approximation to the quantity

$$(6.2) \quad G_i = \frac{d}{d\Omega} j_0(\Omega) \cdot V_i,$$

where V_i is the i th vector of the basis (V_i) of W_N .

As the domain Ω is triangulated, and approximate numerical values of V_i , p_Ω , and p_Ω^* are provided by the FEM solver, it is possible to compute a numerical approximation of the G_i 's by means of any quadrature formula that accurately approximates volume or surface integrals.

REMARK 6.1. *It is shown in [14] that the order of precision of the approximate value of G_i is the same as the FEM used to compute the direct and adjoint states.*

7. The results. Before starting the optimization computations, several tests and validations must be done. We give below some methods for the validation of each step in the complete optimization process.

7.1. Validation of an optimization program. The validation of the FEM solver which computes the state variables is classical: it is done by comparing the computed solutions to analytical ones when available, or by means of evaluating the convergence rate for refined meshes. Furthermore, in some cases, standard benchmarks must be passed.

The gradient or sensitivity validation is less usual: in some particular cases, there are analytical results to compare with the calculated DCG. But, in general cases, it is necessary to compare the DCG values with a finite difference approximation.

The method is the following:

- choose a master node number $\mathbf{i}$;
- compute the vector field $V_{\mathbf{i}}$ using (6.1);
- choose a small perturbation parameter $t \ll 1$;
- compute a perturbed domain $\Omega_t^+ = \Omega + t.V_{\mathbf{i}}$;
- evaluate the cost $j_0(\Omega_t^+)$ (call the FEM solver);
- compute a perturbed domain $\Omega_t^- = \Omega - t.V_{\mathbf{i}}$;
- evaluate the cost $j_0(\Omega_t^-)$ (call the FEM solver).

The finite difference approximation (FDA) is then

$$\text{FDA}(\mathbf{i}) = \frac{j_0(\Omega_t^+) - j_0(\Omega_t^-)}{2t}.$$

Finally, compare $\text{FDA}(\mathbf{i})$ with the approximation $\text{DCG}(\mathbf{i})$.

Numerical experiments have shown that first-order finite difference approximation is not accurate enough. The parameter t must be small enough to ensure the consistency of the FDA and large enough to make the perturbation detectable by the FEM solver.

The last step is the validation of the computed optimum.

Since we do not know whether the cost function is convex, coercive, etc., with respect to the domain variable, the stopping criteria implemented in the minimization subroutines only ensure that, at convergence, the iterate is only a critical point.

One method of validation of the approximate optimum is the following:

- run the optimization process until the stopping criteria (small gradients, small cost variation, etc.) are fulfilled; the obtained shape is then numerically a critical point;
- perturb the critical shape with respect to the master nodes, and observe that the cost increases in all directions; the critical shape is then a local minimum;
- start from different initial shapes, and observe that the local minimum is always the same; the local minimum is then a global and unique one. This step is not always successful, especially when the minimum is not unique (we then have a numerical proof of the nonuniqueness of the optimum).

7.2. The model. First, we describe the fixed data.

The domain Ω is the unit square, with hard-sound boundaries. The acoustical source is a Dirac pointwise distribution, with origin located at (0.9,0.5) and of magnitude 1. The sensitive area Z is a rectangular zone, as represented in Figure 7.1. The target element, i.e., the design variable, is the left side of the square, approximated by cubic splines. The master nodes are allowed to move horizontally with a box constraint, i.e., all the abscissa x_i of the nodes fulfill the condition $-0.05 \leq x_i \leq 0.05$.

This model configuration may correspond, for instance, to the situation where a machine user is in front of a central processor unit box, which is made of sound-hard material (for the comfort of the user). The fan fixed at the back side generates harmonic acoustical waves that may have a negative impact on the computer performance. We then seek front-side geometries that minimize the effects of the fan on the electronic chips (this is a simplified approach; in fact, a pertinent model must take into account at least the coupled vibroacoustic behavior).

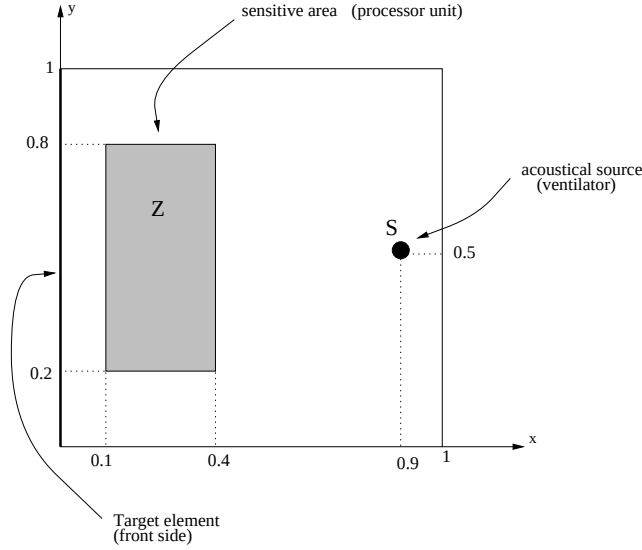


FIG. 7.1. Description of the model configuration.

 TABLE 7.1
 Comparison between DCG and FDA for $k = 1$.

Mesh	j_0	DCG	FDA
441	1.36728	1.17916	1.17047
1681	1.37730	1.19309	1.19088
3721	1.38027	1.19817	1.19847

The optimization computations are done for

- different wave numbers: $k_1 = 1$, $k_2 = 4$, $k_3 = 10$;
- different cost functions:

$$\begin{aligned}
 j_2(\Omega) &= \int_Z |p_\Omega(x)|^2 dZ, \\
 j_{10}(\Omega) &= \left(\int_Z |p_\Omega(x)|^{10} dZ \right)^{1/10}, \\
 j_\infty(\Omega) &= \max_{x \in Z} |p_\Omega(x)|^2.
 \end{aligned}$$

7.3. The sensitivity tests. The sensitivity tests are done for the cost function j_0 . The data are

- the fixed model configuration described above, with one master node located at $(0.5, 0.0)$;
- the cost function: $j_0(\Omega) = |p_\Omega(x_0)|^2$ where $x_0 = (0.15, 0.25)$;
- the wave numbers: $k = 1$ and $k = 4$.

Tables 7.1 ($k = 1$) and 7.2 ($k = 4$) compare the DCG of j_0 to the FDA obtained for different meshes of the unit square.

REMARK 7.1. *If we consider the values of the cost j_0 and the gradient DCG obtained with the finest mesh (3712 nodes) as exact values, then the convergence ratio r_c for the cost is close to the convergence ratio r_g for the gradient:*

- case $k = 1$; $r_c = 4.3$, $r_g = 3.6$;
- case $k = 4$; $r_c = 4.0$, $r_g = 3.8$.

The above results corroborate Remark 6.1.

TABLE 7.2
Comparison between DCG and FDA for $k = 4$.

Mesh	j_0	DCG	FDA
441	0.00648	0.35443	0.34758
1681	0.00710	0.39362	0.39297
3721	0.00731	0.40768	0.40843

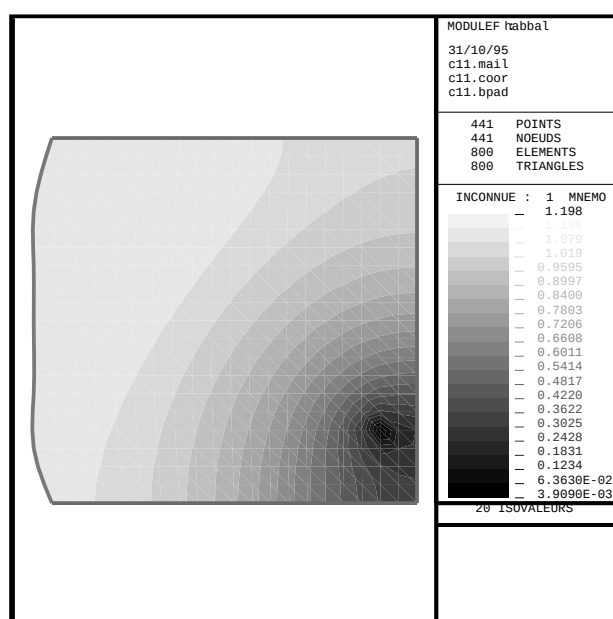


FIG. 7.2. Optimal geometry and pressure distribution for $k = 1$.

7.4. Optimization results. We present below some optimization results obtained for the model problem presented above. The source location is now at $(0.9, 0.2)$.

The minimization of the nonsmooth cost j_∞ was done using the subroutine M2FC1, based on the algorithm presented in [9], developed at the MODULOPT project (INRIA, France). Other NSO codes, based on more recent algorithms such as the BT method (see [19]) or NOA (see [20]), could be competitive with M1FC1.

7.4.1. Results for $k = 1$ and $k = 4$. The optimal shapes are the same for the three costs: j_2 , j_{10} , and j_∞ . They are obtained by the projection on the bounds -0.05 and $+0.05$. See Figures 7.2 and 7.3.

For the case $k = 1$, this could be explained as follows. Since the sensitive area Z is fixed, the optimization process tends to make the left side as distant as possible from Z , in order to minimize the scattering effect.

7.4.2. Results for $k = 10$. Figure 7.4 shows the initial shape and pressure distribution.

Figures 7.5, 7.6, and 7.7 show both the optimal geometry and the optimal acoustical pressure for the three considered costs. We observe that the optimal shapes are slightly different from one cost to another.

Table 7.3 shows the initial and optimal values for the different costs, and the maximal pressure in the three cases.

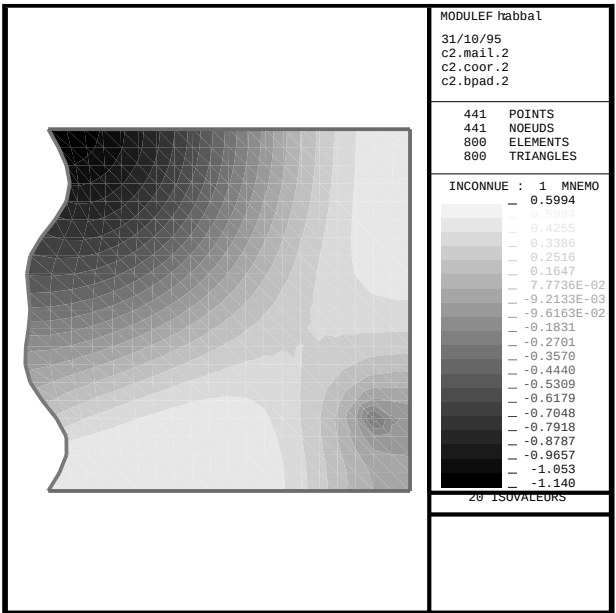


FIG. 7.3. Optimal geometry and pressure distribution for $k = 4$.

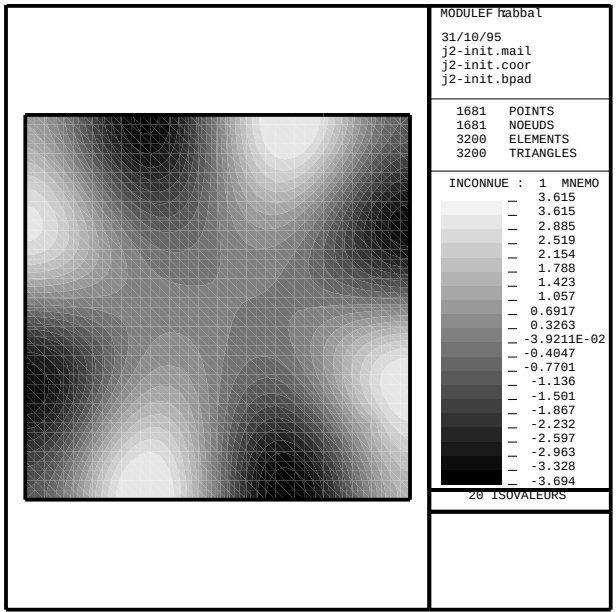


FIG. 7.4. Initial geometry and pressure distribution.

8. Conclusion. The numerical experiments show that for low frequencies (50–200 Hz) the optimal design problem leads to more or less trivial optimal solutions. The acoustical pressure distribution seems to be insensitive to small changes in the shape.

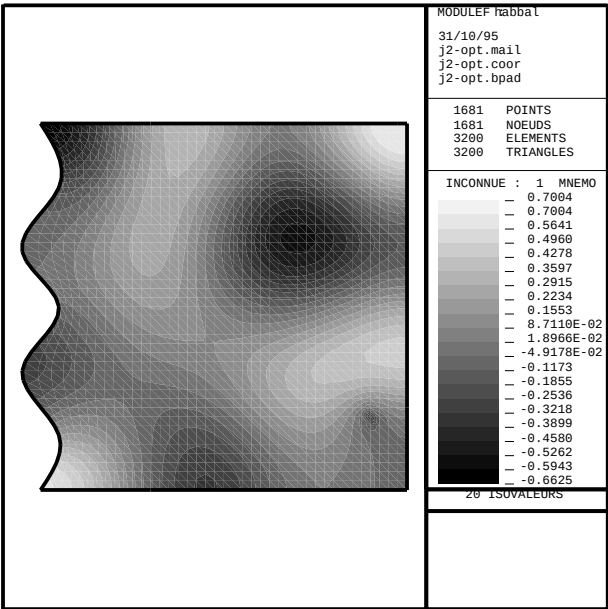


FIG. 7.5. Optimal geometry and pressure distribution for the cost j_2 .

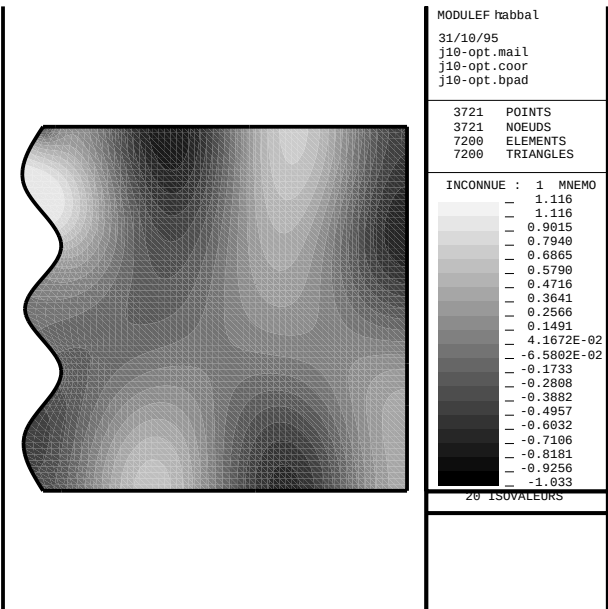


FIG. 7.6. Optimal geometry and pressure distribution for the cost j_{10} .

For large enough (medium) frequencies (550 Hz), the results are more interesting. The optimized boundaries show some kind of oscillating behavior, and the ratio between the initial and optimal costs is very important.

In the case of absorbing material, the oscillating effect increases the absorbed part of the acoustic energy, which could lead to a greater ratio of initial to optimal cost than in the total reflection case.

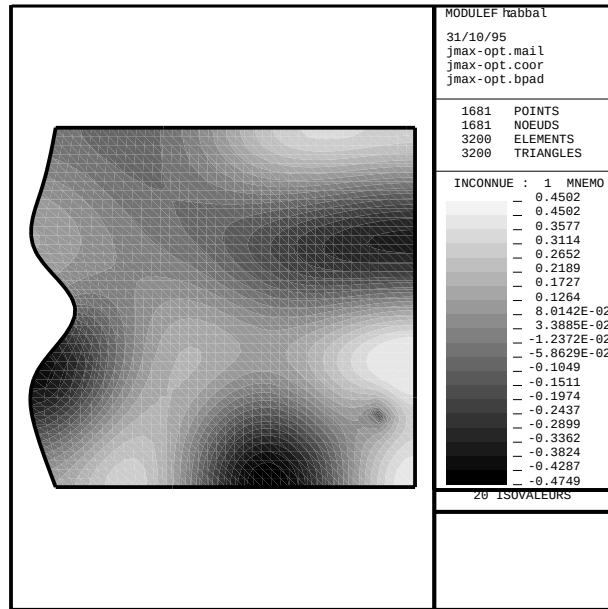

 FIG. 7.7. Optimal geometry and pressure distribution for the cost j_∞ .

 TABLE 7.3
 Initial-optimal values and maximal pressure for the different cost functions.

Costs	j_2	j_{10}	j_∞
initial	0.18623	3.19572	5.54215
optimal	0.00349	0.37819	0.03230
p_{max}	0.27162	0.67023	0.19498

However, it is important to remark that the optimal shapes strongly depend on the wave frequency. One solution to this problem consists of considering as the objective function the maximum of pressure in the sensitive area, over a wide frequency band.

Finally, let us remark that the mathematical results presented here are still valid for unbounded domains (one has local regularity, and it is sufficient to obtain the same results as for bounded domains).

REFERENCES

- [1] P. A. NELSON AND S. J. ELLIOT, *Active Control of Sound*, Academic Press, New York, 1992.
- [2] D. DUHAMEL, *L'acoustique des problèmes couplés fluide-structure. application au contrôle actif du son*, Thèse de Doctorat, Ecole Nationale des Ponts et Chaussées, Paris, France, 1994.
- [3] A. MYSLINSKI AND J. SOKOLOWSKI, *Nondifferentiable optimization problems for elliptic systems*, SIAM J. Control Optim., 23 (1985), pp. 632–648.
- [4] H. J.-P. MORAND AND R. OHAYON, *Interactions Fluides-Structures*, RMA 23, Masson, Paris, France, 1992.
- [5] N. S. TRUDINGER AND D. GILBARG, *Elliptic Partial Differential Equations of Second Order*, Springer-Verlag, Berlin, Heidelberg, New York, 1977.
- [6] B. ROUSSELET, *Quelques résultats en optimisation de domaines*, Thèse de Doctorat d'Etat, Université de Nice–Sophia Antipolis, Nice, France, 1982.
- [7] C. LEMARÉCHAL, *Nondifferentiable Optimization*, Handbooks Oper. Res. Management Sci., North-Holland, Amsterdam, The Netherlands, 1989.

- [8] J. B. HIRIART-URRUTY AND C. LEMARÉCHAL, *Convex Analysis and Minimization Algorithms I and II*, Springer-Verlag, Berlin, Heidelberg, 1993.
- [9] J. S. STRODIOT, C. LEMARÉCHAL, AND A. BIHAIN, *On a bundle algorithm for nonsmooth optimization*, in *Nonlinear Programming 4*, O. L. Mangasarian, G. L. Meyer, and S. M. Robinson, eds., Academic Press, New York, 1981, pp. 245–282.
- [10] F. H. CLARKE, *Optimization and Nonsmooth Analysis*, Wiley, New York, 1983.
- [11] M. M. MAKELA AND P. NEITTAANMAKI, *Nonsmooth Optimization: Analysis and Algorithms with Applications to Optimal Control*, World Scientific, Singapore, 1992.
- [12] C. LEMARÉCHAL, *Nondifferentiable optimization*, in *Nonlinear Optimization*, L. C. W. Dixon, E. Spedicato, and G. P. Szego, eds., Birkhauser, Basel, 1980, pp. 149–199.
- [13] J. CEA, *Conception optimale ou identification de formes, calcul rapide de la dérivée directionnelle de la fonction coût*, MAA Notes, 20 (1986), pp. 371–402.
- [14] M. MASMOUDI, *Outils pour la Conception Optimale de Forme*, Thèse d'Etat, University of Nice, Nice, France, 1987.
- [15] A. HABBAL, *Theoretical and numerical study of nonsmooth shape optimization applied to the arch problem*, Mech. Structures Mach., 20 (1992), pp. 93–117.
- [16] J. NECAS, *Les Méthodes Directes en Théorie des Equations Elliptiques*, Masson, Paris, France, 1967.
- [17] J. SIMON AND F. MURAT, *Sur le Contrôle par un Domaine Géométrique*, Thèse d'Etat, Université Paris 6, Paris, France, 1976.
- [18] O. PIRONNEAU, *Optimal Shape Design for Elliptic Systems*, Springer-Verlag, Berlin, Heidelberg, New York, 1984.
- [19] H. SCHRAMM AND J. ZOWE, *A version of the bundle idea for minimizing a nonsmooth function: Conceptual idea, convergence analysis, numerical results*, SIAM J. Optim., 2 (1992), pp. 121–152.
- [20] K. C. KIWIEL, *Proximity control in bundle methods for convex nondifferentiable optimization*, Math. Programming, 46 (1990), pp. 105–122.

A SUPERLINEARLY CONVERGENT PRIMAL-DUAL INFEASIBLE-INTERIOR-POINT ALGORITHM FOR SEMIDEFINITE PROGRAMMING*

FLORIAN A. POTRA[†] AND RONGQIN SHENG[†]

Abstract. A primal-dual infeasible-interior-point path-following algorithm is proposed for solving semidefinite programming (SDP) problems. If the problem has a solution, then the algorithm is globally convergent. If the starting point is feasible or close to being feasible, the algorithm finds an optimal solution in at most $O(\sqrt{n}L)$ iterations, where n is the size of the problem and L is the logarithm of the ratio of the initial error and the tolerance. If the starting point is large enough, then the algorithm terminates in at most $O(nL)$ steps either by finding a solution or by determining that the primal-dual problem has no solution of norm less than a given number. Moreover, we propose a sufficient condition for the superlinear convergence of the algorithm. In addition, we give two special cases of SDP for which the algorithm is quadratically convergent.

Key words. semidefinite programming, path-following, infeasible-interior-point algorithm, polynomiality, superlinear convergence

AMS subject classifications. 65K05, 90C25, 90C30

PII. S1052623495294955

1. Introduction. In this paper we consider the SDP problem

$$(1.1) \quad (P) \quad \min\{C \bullet X : A_i \bullet X = b_i, i = 1, \dots, m, X \succeq 0\},$$

and its associated dual problem

$$(1.2) \quad (D) \quad \max \left\{ b^T y : \sum_{i=1}^m y_i A_i + S = C, S \succeq 0 \right\},$$

where $C \in \mathcal{S}^n$, $A_i \in \mathcal{S}^n$, $i = 1, \dots, m$, $b = (b_1, \dots, b_m)^T \in \mathbb{R}^m$ are given data and $X \in \mathcal{S}_+^n$, $(y, S) \in \mathbb{R}^m \times \mathcal{S}_+^n$ are the primal and dual variables, respectively. Here we use $\mathcal{S}^n$ to denote the set of all $n \times n$ symmetric matrices and $\mathcal{S}_+^n$ the set of all $n \times n$ symmetric positive semidefinite matrices. By $G \bullet H$ we denote the trace of $(G^T H)$. For simplicity we assume that A_i , $i = 1, \dots, m$, are linearly independent.

Throughout this paper we assume that both (1.1) and (1.2) have finite solutions and that their optimal values are equal. Under these assumptions, X^* and (y^*, S^*) are solutions of (1.1) and (1.2) if and only if they are solutions of the following nonlinear system:

$$(1.3a) \quad A_i \bullet X = b_i, i = 1, \dots, m,$$

$$(1.3b) \quad \sum_{i=1}^m y_i A_i + S = C,$$

$$(1.3c) \quad X \bullet S = 0, X \succeq 0, S \succeq 0.$$

*Received by the editors November 20, 1995; accepted for publication (in revised form) October 14, 1997; published electronically September 23, 1998. This work was supported in part by NSF grant DMS 9305760.

<http://www.siam.org/journals/siopt/8-4/29495.html>

[†]Department of Mathematics, The University of Iowa, Iowa City, IA 52242 (potra@math.uiowa.edu, rsheng@math.uiowa.edu).

Some primal-dual interior-point methods for linear programming have been successfully extended to solve the SDP problems (1.3). For a survey of results obtained before 1993 in this field see the paper of Alizadeh [1]. More recent results can be found in [4, 2, 3, 7, 9, 16].

Kojima, Shindoh, and Hara [9], Nesterov and Todd [13], and Monteiro [12] extended some interior-point methods for linear programming (LP) to SDP. In the latter paper Monteiro developed a new formulation of the primal-dual search direction originally introduced in [9]. All above-mentioned methods, with the exception of the infeasible-interior-point potential-reduction method of Kojima, Shindoh, and Hara [9], require a strictly feasible starting point and therefore are *feasible-interior-point methods*. More recently, Zhang [17], Kojima, Shida, and Shindoh [8], and the present authors (in the first version of this paper) independently proposed new infeasible-interior-point path-following algorithms for SDP. In this version, we have corrected some flaws in the local convergence analysis contained in the first version. Our algorithm is a predictor-corrector method generalizing the interior-point method for linear programming proposed by Mizuno, Todd, and Ye [11]. We note that the algorithm of [11] has been also generalized for linear complementarity problems with feasible starting points in [6] and with infeasible starting points in [10, 15]. We also mention that the Mizuno–Todd–Ye predictor-corrector method has been extended to self-scaled cones, which include SDP, by Nesterov and Todd [13] under the assumption that the starting point is strictly feasible.

The algorithm to be presented in this paper is globally convergent whenever the problem (1.3) has a solution. The starting point does not have to be feasible. In particular we can take as a starting point any positive multiple of the identity matrix.

If the starting point is feasible or close to being feasible our algorithm finds a solution in at most $O(\sqrt{n}L)$ iterations. If the starting point is large enough, then the algorithm terminates in at most $O(nL)$ steps either by finding a solution or by determining that the primal-dual problem has no solution of norm less than a given number. Moreover, we propose a sufficient condition for the superlinear convergence of the algorithm. The sufficient condition is satisfied under conditions (A), (B), and (C) of Kojima, Shida, and Shindoh [8]. In addition, we give two special cases of SDP for which the algorithm is quadratically convergent. Superlinearly convergent algorithms tend to perform in practice much better than indicated by their iteration complexity, which is based on global linear convergence estimates. Indeed, superlinear convergence has been observed experimentally in efficient practical algorithms [14].

The following notation and terminology are used throughout the paper:

- $\mathbb{R}^p$: the p -dimensional Euclidean space;
- $\mathbb{R}_+^p$: the nonnegative orthant of $\mathbb{R}^p$;
- $\mathbb{R}_{++}^p$: the positive orthant of $\mathbb{R}^p$;
- $\mathbb{R}^{p \times q}$: the set of all $p \times q$ matrices with real entries;
- $\mathcal{S}^p$: the set of all $p \times p$ symmetric matrices;
- $\mathcal{S}_+^p$: the set of all $p \times p$ symmetric positive semidefinite matrices;
- $\mathcal{S}_{++}^p$: the set of all $p \times p$ symmetric positive matrices;
- $[M]_{ij}$: the (i, j) th entry of a matrix M ;
- $\text{Tr}(M)$: the trace of a $p \times p$ matrix, equals $\sum_{i=1}^p [M]_{ii}$;
- $M \succeq 0$: M is positive semidefinite;
- $M \succ 0$: M is positive definite;
- $\lambda_i(M)$, $i = 1, \dots, n$: the eigenvalues of $M \in \mathcal{S}^n$;
- $\lambda_{\max}(M)$, $\lambda_{\min}(M)$: the largest and smallest eigenvalues of $M \in \mathcal{S}^n$;

$G \bullet H \equiv \text{Tr}(G^T H)$;
 $\|\cdot\|$: Euclidean norm of a vector and the corresponding norm of a matrix, i.e.,
 $\|y\| \equiv \sqrt{\sum_{i=1}^p y_i^2}$, $\|M\| \equiv \max\{\|My\| : \|y\| = 1\}$;
 $\|M\|_F \equiv \sqrt{M \bullet M}$, $M \in \mathbb{R}^{p \times q}$: Frobenius norm of a matrix;
 $\|(G, H)\|_F \equiv \sqrt{G \bullet G + H \bullet H}$, $G, H \in \mathbb{R}^{p \times q}$;
 $M^k = o(1)$: $\|M^k\| \rightarrow 0$ as $k \rightarrow \infty$;
 $M^k = O(1)$: $\|M^k\|$ is bounded;
 $M^k = o(\nu_k)$: $M^k/\nu_k = o(1)$;
 $M^k = O(\nu_k)$: $M^k/\nu_k = O(1)$.

2. An infeasible-interior-point algorithm. We denote the feasible set of the problem (1.3) by

$$\mathcal{F} = \{(X, y, S) \in \mathcal{S}_+^n \times \mathbb{R}^m \times \mathcal{S}_+^n : (X, y, S) \text{ satisfies (1.3a) and (1.3b)}\}$$

and its solution set by

$$\mathcal{F}^* = \{(X, y, S) \in \mathcal{F} : X \bullet S = 0\}.$$

The residues of (1.3a) and (1.3b) are denoted by

$$(2.1a) \quad r_i = b_i - A_i \bullet X, \quad i = 1, \dots, m,$$

$$(2.1b) \quad R_d = C - \sum_{i=1}^m y_i A_i - S.$$

For any given $\epsilon > 0$ we define the set of ϵ -approximate solutions of (1.3) as

$$\mathcal{F}_\epsilon = \{Z = (X, y, S) \in \mathcal{S}_+^n \times \mathbb{R}^m \times \mathcal{S}_+^n : \\ X \bullet S \leq \epsilon, \quad |r_i| \leq \epsilon, \quad i = 1, \dots, m, \quad \|R_d\| \leq \epsilon\}.$$

In what follows we present an algorithm that finds a point in this set in a finite number of steps (provided the problem has a solution). The algorithm will perform in a neighborhood of the infeasible central path starting at

$$\begin{aligned} (X^0, y^0, S^0) &\in \mathcal{S}_{++}^n \times \mathbb{R}^m \times \mathcal{S}_{++}^n : \\ \mathcal{C}(X^0, y^0, S^0) &= \{Z = (X, y, S) \in \mathcal{S}_{++}^n \times \mathbb{R}^m \times \mathcal{S}_{++}^n : \\ &\quad XS = \tau I, \quad r_i = (\tau/\tau_0)r_i^0, \quad i = 1, \dots, m, \quad R_d = (\tau/\tau_0)R_d^0\}, \end{aligned}$$

where R_d^0 , r_i^0 , $i = 1, \dots, m$, are residues at (X^0, y^0, S^0) which satisfy $X^0 S^0 = \tau_0 I$. In our algorithm the positive parameter τ will be driven to zero and therefore the residues will also be driven to zero at the same rate as τ . We use the following neighborhood of the above central path:

$$\begin{aligned} \mathcal{N}(\gamma, \tau) &= \{(X, y, S) \in \mathcal{S}_{++}^n \times \mathbb{R}^m \times \mathcal{S}_{++}^n : \|X^{1/2} S X^{1/2} - \tau I\|_F \leq \gamma \tau\} \\ &= \left\{ (X, y, S) \in \mathcal{S}_{++}^n \times \mathbb{R}^m \times \mathcal{S}_{++}^n : \left(\sum_{i=1}^m (\lambda_i(XS) - \tau)^2 \right)^{1/2} \leq \gamma \tau \right\}, \end{aligned}$$

where γ is a constant such that $0 < \gamma < 1$. Throughout the paper we also use the notation

$$(2.2) \quad \mu = (X \bullet S)/n.$$

The algorithm depends on two positive parameters α, β satisfying the inequalities

$$(2.3) \quad \frac{\beta^2}{2(1-\beta)^2} \leq \alpha < \beta < \frac{\beta}{1-\beta} < 1.$$

For example, $\alpha = 0.25$, $\beta = 0.41$ satisfy (2.3).

At a typical step of our algorithm we are given $(X, y, S) \in \mathcal{N}(\alpha, \tau)$ and obtain a predictor direction $(U, w, V) \in \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n$ by solving the linear system

$$(2.4a) \quad X^{-1/2}(XV + US)X^{1/2} + X^{1/2}(VX + SU)X^{-1/2} = -2X^{1/2}SX^{1/2},$$

$$(2.4b) \quad A_i \bullet U = r_i, \quad i = 1, \dots, m,$$

$$(2.4c) \quad \sum_{i=1}^m w_i A_i + V = R_d.$$

We will see later on that the above linear system has a unique solution, which we call the affine scaling direction. If we take a step length θ along this direction we obtain the points

$$X(\theta) = X + \theta U, \quad y(\theta) = y + \theta w, \quad S(\theta) = S + \theta V.$$

Theoretically we would like to compute the step length

$$(2.5) \quad \check{\theta} = \max \left\{ \tilde{\theta} \in [0, 1] : \left(\sum_{i=1}^n (\lambda_i(X(\theta)S(\theta)) - (1-\theta)\tau)^2 \right)^{1/2} \leq \beta(1-\theta)\tau \quad \forall \theta \in [0, \tilde{\theta}] \right\}.$$

However, this involves computing the root of a complicated nonlinear equation. In Lemma 2.5 we will show that

$$(2.6) \quad \check{\theta} \geq \hat{\theta},$$

where

$$(2.7) \quad \hat{\theta} \equiv \frac{2}{\sqrt{1 + 4\delta/(\beta - \alpha)} + 1}$$

and

$$(2.8) \quad \delta \equiv \frac{1}{\tau} \|X^{-1/2}UVX^{1/2}\|_F.$$

In what follows we assume that a step length $\bar{\theta}$ satisfying

$$(2.9) \quad \check{\theta} \geq \bar{\theta} \geq \hat{\theta}$$

is computed, and we consider the predicted points

$$(2.10) \quad \bar{X} = X + \bar{\theta}U, \quad \bar{y} = y + \bar{\theta}w, \quad \bar{S} = S + \bar{\theta}V.$$

In the case of $\bar{\theta} = 1$ (which is very unlikely), it is easily seen that a solution $(\bar{X}, \bar{y}, \bar{S}) \in \mathcal{F}^*$ is at hand and the algorithm terminates. Now suppose that $\bar{\theta} < 1$. Then $\bar{X}$

and $\bar{S}$ are symmetric positive definite matrices, since $(X(\theta), y(\theta), S(\theta)) \in \mathcal{N}(\beta, (1 - \theta)\tau) \ \forall \theta \in [0, \bar{\theta}]$. Therefore we can define the corrector direction $(\bar{U}, \bar{w}, \bar{V})$ as the solution of the following linear system:

$$(2.11a) \quad \bar{X}^{-1/2}(\bar{X}\bar{V} + \bar{U}\bar{S})\bar{X}^{1/2} + \bar{X}^{1/2}(\bar{V}\bar{X} + \bar{S}\bar{U})\bar{X}^{-1/2} \\ = 2[(1 - \bar{\theta})\tau I - \bar{X}^{1/2}\bar{S}\bar{X}^{1/2}],$$

$$(2.11b) \quad A_i \bullet \bar{U} = 0, \quad i = 1, \dots, m,$$

$$(2.11c) \quad \sum_{i=1}^m \bar{w}_i A_i + \bar{V} = 0.$$

We will prove later on that the above linear system has a unique solution. By taking a unit step length along the corrector direction we obtain a new point

$$(2.12) \quad X^+ = \bar{X} + \bar{U}, \quad y^+ = \bar{y} + \bar{w}, \quad S^+ = \bar{S} + \bar{V}.$$

Clearly

$$(2.13) \quad r_i^+ = (1 - \bar{\theta})r_i, \quad i = 1, \dots, m, \quad R_d^+ = (1 - \bar{\theta})R_d.$$

Correspondingly, we define

$$(2.14) \quad \tau_+ = (1 - \bar{\theta})\tau.$$

Let us note that in setting up the linear systems (2.4) and (2.11) it is not necessary to compute square roots of matrices. Indeed, as pointed out by Monteiro [12], it is easily seen that any equation of the form

$$X^{-1/2}(XV + US)X^{1/2} + X^{1/2}(VX + SU)X^{-1/2} = 2(\sigma I - X^{1/2}SX^{1/2})$$

can be written equivalently in the form

$$2XVX + USX + SXU = 2(\sigma X - XSX).$$

Summarizing, we can formally define our algorithm as follows.

ALGORITHM 2.1. Choose $(X^0, y^0, S^0) \in \mathcal{N}(\alpha, \tau_0)$ with $\tau_0 = \mu_0 = (X^0 \bullet S^0)/n$ and set $\psi_0 = 1$. For $k = 0, 1, \dots$, do A1 through A5:

- A1. Set $X = X^k$, $y = y^k$, $S = S^k$, $\psi = \psi_k$, $\tau = \tau_k$, and define $R_d = C - \sum_{i=1}^m y_i A_i - S$, $r_i = b_i - A_i \bullet X$, $i = 1, \dots, m$.
- A2. If $\max\{X \bullet S, \|R_d\|, |r_i|, i = 1, \dots, m\} \leq \epsilon$, then report $(X, y, S) \in \mathcal{F}_\epsilon$ and terminate.
- A3. Find the solution U, w, V of the linear system (2.4), define $\bar{X}, \bar{y}, \bar{S}$ as in (2.10), and set $\psi_+ = (1 - \bar{\theta})\psi$ for a $\bar{\theta}$ satisfying (2.9). If $\bar{\theta} = 1$, then report $(\bar{X}, \bar{y}, \bar{S}) \in \mathcal{F}^*$ and terminate.
- A4. Find the solution $\bar{U}, \bar{w}, \bar{V}$ of the linear system (2.11) and define X^+, y^+, S^+ and τ_+ as in (2.12) and (2.14).
- A5. Set $X^{k+1} = X^+$, $S^{k+1} = S^+$, $\tau_{k+1} = \tau_+$, $\bar{\theta}_k = \bar{\theta}$, $\psi_{k+1} = \psi_+$, $R_d^k = R_d$, $r_i^k = r_i$, $i = 1, \dots, m$.

To analyze the algorithm we need the following technical results.

LEMMA 2.2. Suppose that $M \in \mathbb{R}^{p \times p}$ is a nonsingular matrix and $E \in \mathbb{R}^{p \times p}$ has only real eigenvalues. Then,

$$(2.15) \quad \lambda_{\max}(E) \leq \frac{1}{2} \lambda_{\max}(MEM^{-1} + (MEM^{-1})^T),$$

$$(2.16) \quad \lambda_{\min}(E) \geq \frac{1}{2} \lambda_{\min}(MEM^{-1} + (MEM^{-1})^T),$$

$$(2.17) \quad \|G\|_F \leq \frac{1}{2} \|MGM^{-1} + (MGM^{-1})^T\|_F \quad \forall G \in \mathcal{S}^p.$$

Proof. See Lemmas 3.2 and 3.3 of Monteiro [12]. $\square$

LEMMA 2.3 (Monteiro [12, Lemma 4.4]). *Let $(X, y, S) \in \mathcal{N}(\gamma, \tau)$ for some $\gamma \in [0, 1]$ and $\tau > 0$. Suppose that $(D_x, \Delta y, D_s) \in \mathbb{R}^{n \times n} \times \mathbb{R}^m \times \mathbb{R}^{n \times n}$ is a solution of the linear system*

$$(2.18a) \quad X^{-1/2}(XD_s + D_xS)X^{1/2} + X^{1/2}(D_sX + SD_x)X^{-1/2} = 2H,$$

$$(2.18b) \quad A_i \bullet D_x = 0, \quad i = 1, \dots, m,$$

$$(2.18c) \quad \sum_{i=1}^m \Delta y_i A_i + D_s = 0$$

for some $H \in \mathbb{R}^{n \times n}$, and let $\delta_x = \tau \|X^{-1/2}D_xX^{-1/2}\|_F$ and $\delta_s = \|X^{1/2}D_sX^{1/2}\|_F$. Then

$$\delta_x \delta_s \leq \frac{1}{2} (\delta_x^2 + \delta_s^2) \leq \frac{\|H\|_F^2}{2(1-\gamma)^2}.$$

The following corollary is essentially Corollary 2.3 of Monteiro [12].

COROLLARY 2.4. *Let $(X, y, S) \in \mathcal{N}(\gamma, \tau)$ for some $\gamma \in [0, 1]$ and $\tau > 0$. Then, for every $H \in \mathcal{S}^n$, the system of linear equations (2.18) has a unique solution $(D_x, \Delta y, D_s) \in \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n$. Therefore, both linear systems (2.4) and (2.11) have unique solutions.*

LEMMA 2.5. *If $(X, y, S) \in \mathcal{N}(\alpha, \tau)$, then $\tilde{\theta} \in [0, 1]$ defined by (2.5) satisfies (2.6) where $\hat{\theta}$ is given by (2.7) and (2.8).*

Proof. By definition, for any $\theta \in \mathbb{R}$, we have

$$(2.19) \quad \begin{aligned} & X(\theta)S(\theta) - (1-\theta)\tau I = (X + \theta U)(S + \theta V) - (1-\theta)\tau I \\ & = (1-\theta)(XS - \tau I) + \theta XS + \theta(XV + US) + \theta^2 UV. \end{aligned}$$

If we set

$$P(\theta) \equiv X^{-1/2}[X(\theta)S(\theta) - (1-\theta)\tau I]X^{1/2},$$

then, in view of (2.19) and (2.4a), we obtain

$$\begin{aligned} P(\theta) + P(\theta)^T &= 2(1-\theta)(X^{1/2}SX^{1/2} - \tau I) \\ &\quad + \theta[X^{-1/2}(XV + US)X^{1/2} + X^{1/2}(VX + SU)X^{-1/2} + 2X^{1/2}SX^{1/2}] \\ &\quad + \theta^2(X^{-1/2}UVX^{1/2} + X^{1/2}VUX^{-1/2}) \\ &= 2(1-\theta)(X^{1/2}SX^{1/2} - \tau I) + \theta^2(X^{-1/2}UVX^{1/2} + X^{1/2}VUX^{-1/2}). \end{aligned}$$

Therefore,

$$(2.20) \quad \begin{aligned} & \frac{1}{2} \|P(\theta) + P(\theta)^T\|_F \\ & \leq (1-\theta) \|X^{1/2}SX^{1/2} - \tau I\|_F + \theta^2 \|X^{-1/2}UVX^{1/2}\|_F \\ & \leq \alpha\tau(1-\theta) + \theta^2\delta\tau. \end{aligned}$$

Hence, if $\widehat{\theta} < 1$, then we must have $X(\theta) \succ 0$, $S(\theta) \succ 0$ for all $\theta \in [0, \widehat{\theta}]$. Otherwise, there must exist $0 \leq \theta' \leq \widehat{\theta} < 1$ such that $X(\theta')S(\theta')$ is singular, which means

$$(2.21) \quad \lambda_{\min}(X(\theta')S(\theta') - (1 - \theta')\tau) \leq -(1 - \theta')\tau.$$

However, using (2.16) with $M = X^{-1/2}$ and $E = X(\theta')S(\theta') - (1 - \theta')\tau$, we have

$$\begin{aligned} & \lambda_{\min}(X(\theta')S(\theta') - (1 - \theta')\tau) \\ & \geq \frac{1}{2}\lambda_{\min}(P(\theta') + P(\theta')^T) \\ & \geq -\frac{1}{2}\|P(\theta') + P(\theta')^T\|_F \\ & \geq -[\alpha\tau(1 - \theta') + (\theta')^2\delta\tau] \quad (\text{from (2.20)}) \\ & \geq -\beta(1 - \theta')\tau, \end{aligned}$$

which contradicts (2.21). Since $X(\theta) \succ 0$, its square root $X(\theta)^{1/2}$ exists and is uniquely defined. Applying (2.17) of Lemma 2.2 with

$$G = X(\theta)^{1/2}S(\theta)X(\theta)^{1/2} - (1 - \theta)\tau I, \quad M = X^{-1/2}X(\theta)^{1/2},$$

and noting that $P(\theta) = MGM^{-1}$, we obtain

$$\begin{aligned} & \left(\sum_{i=1}^n (\lambda_i(X(\theta)S(\theta)) - (1 - \theta)\tau)^2 \right)^{1/2} \\ & = \|X(\theta)^{1/2}S(\theta)X(\theta)^{1/2} - (1 - \theta)\tau I\|_F \\ & \leq \frac{1}{2}\|P(\theta) + P(\theta)^T\|_F \\ & \leq \alpha\tau(1 - \theta) + \theta^2\delta\tau \quad (\text{from (2.20)}) \\ & \leq \beta(1 - \theta)\tau. \end{aligned}$$

Therefore, $(X(\theta), y(\theta), S(\theta)) \in \mathcal{N}(\beta, (1 - \theta)\tau) \forall \theta \in [0, \widehat{\theta}]$. Thus, if $\widehat{\theta} < 1$, then (2.6) is satisfied. Similarly, if $\widehat{\theta} = 1$, then we must have $(X(\theta), y(\theta), S(\theta)) \in \mathcal{N}(\beta, (1 - \theta)\tau) \forall \theta \in [0, 1)$, which implies $X(1) \succeq 0$, $S(1) \succeq 0$, and $X(1)S(1) = 0$, and therefore $\check{\theta} = 1 = \widehat{\theta}$. $\square$

Before stating our main result let us note that the standard choice of starting points

$$X^0 = \rho_p I, \quad y^0 = 0, \quad S^0 = \rho_d I$$

is perfectly centered and satisfies $(X^0, y^0, S^0) \in \mathcal{N}(\alpha, \tau_0)$, as required in the algorithm.

We will see that if the problem has a solution, then for any $\epsilon > 0$ Algorithm 2.1 terminates in a finite number (say K_ϵ) of iterations. If $\epsilon = 0$, then the algorithm is likely to generate an infinite sequence. However, it may happen that at a certain iteration (let us say at iteration K_0) we have $\bar{\theta} = 1$, which implies that an exact solution is obtained, and therefore the algorithm terminates at iteration K_0 . If this (unlikely) phenomenon does not happen, we set $K_0 = \infty$.

THEOREM 2.6. *For any integer $0 \leq k < K_0$, Algorithm 2.1 defines a triple*

$$(2.22) \quad (X^k, y^k, S^k) \in \mathcal{N}(\alpha, \tau_k),$$

and the corresponding residuals satisfy

$$(2.23) \quad R_d^k = \psi_k R_d^0, \quad r_i^k = \psi_k r_i^0, \quad i = 1, \dots, m,$$

$$(2.24) \quad \tau_k = \psi_k \tau_0,$$

$$(2.25) \quad (1 - \alpha)\tau_k \leq \mu_k = (X^k \bullet S^k)/n \leq (1 + \alpha)\tau_k,$$

where

$$\psi_0 = 1, \quad \psi_k = \prod_{j=0}^{k-1} (1 - \bar{\theta}_j),$$

and $\bar{\theta}_j$ is defined by (2.9).

Proof. The proof is by induction. For $k = 0$, (2.22)–(2.25) are clearly satisfied. Suppose they are satisfied for some $k \geq 0$. As in Algorithm 2.1 we will omit the index k . Therefore we can write

$$\begin{aligned} (X, y, S) &\in \mathcal{N}(\alpha, \tau), \\ R_d &= \psi R_d^0, \quad r_i = \psi r_i^0, \quad i = 1, \dots, m, \\ \tau &= \psi \tau_0, \quad (1 - \alpha)\tau \leq \mu \leq (1 + \alpha)\tau. \end{aligned}$$

The fact that (2.23) and (2.24) hold for $k + 1$ follows immediately from (2.13) and (2.14). From (2.12) and (2.14) we have

$$(2.26) \quad \begin{aligned} X^+ S^+ - \tau_+ I &= (\bar{X} + \bar{U})(\bar{S} + \bar{V}) - \tau_+ I \\ &= \bar{X}\bar{S} - (1 - \bar{\theta})\tau I + \bar{X}\bar{U} + \bar{U}\bar{S} + \bar{U}\bar{V}. \end{aligned}$$

Define

$$P = \bar{X}^{-1/2}(X^+ S^+ - \tau_+ I)\bar{X}^{1/2}.$$

Then, recalling (2.26) and (2.11a), we obtain

$$(2.27) \quad \begin{aligned} P + P^T &= 2[\bar{X}^{1/2}\bar{S}\bar{X}^{1/2} - (1 - \bar{\theta})\tau I] \\ &\quad + [X^{-1/2}(\bar{X}\bar{U} + \bar{U}\bar{S})X^{1/2} + X^{1/2}(\bar{U}\bar{X} + \bar{S}\bar{U})X^{-1/2}] \\ &\quad + [X^{-1/2}\bar{U}\bar{V}X^{1/2} + X^{1/2}\bar{V}\bar{U}X^{-1/2}] \\ &= X^{-1/2}\bar{U}\bar{V}X^{1/2} + X^{1/2}\bar{V}\bar{U}X^{-1/2}. \end{aligned}$$

Since $k < K_0$, we see that $\bar{\theta} < 1$ and that $\bar{X} \succ 0$, $\bar{S} \succ 0$. Hence by applying Lemma 2.3 we deduce

$$(2.28) \quad \begin{aligned} &\|\bar{X}^{-1/2}\bar{U}\bar{X}^{-1/2}\|_F \|\bar{X}^{1/2}\bar{V}\bar{X}^{1/2}\|_F \\ &\leq \frac{\|\bar{X}^{1/2}\bar{S}\bar{X}^{1/2} - (1 - \bar{\theta})\tau I\|_F^2}{2(1 - \beta)^2(1 - \bar{\theta})\tau} \\ &\leq \frac{\beta^2}{2(1 - \beta)^2}(1 - \bar{\theta})\tau \leq \alpha(1 - \bar{\theta})\tau. \end{aligned}$$

Using Lemma 2.3 again, we have

$$\begin{aligned} & \|\bar{X}^{-1/2} \bar{U} \bar{X}^{-1/2}\|_F \\ & \leq \frac{\|\bar{X}^{1/2} \bar{S} \bar{X}^{1/2} - (1 - \bar{\theta}) \tau I\|_F}{(1 - \beta)(1 - \bar{\theta}) \tau} \leq \frac{\beta}{1 - \beta} < 1, \end{aligned}$$

which implies that $I + \bar{X}^{-1/2} \bar{U} \bar{X}^{-1/2} \succ 0$, and therefore $X^+ = \bar{X} + \bar{U} \succ 0$. Thus $(X^+)^{1/2}$ exists. Using (2.26), (2.27), applying Lemma 2.2 with

$$G = (X^+)^{1/2} S^+ (X^+)^{1/2} - \tau_+ I, \quad M = \bar{X}^{-1/2} (X^+)^{1/2},$$

and noting that $P = MGM^{-1}$, we have

$$\begin{aligned} & \|(X^+)^{1/2} S^+ (X^+)^{1/2} - \tau_+ I\|_F \leq \frac{1}{2} \|P + P^T\|_F \\ & = \frac{1}{2} \|\bar{X}^{-1/2} \bar{U} \bar{V} \bar{X}^{1/2} + \bar{X}^{1/2} \bar{V} \bar{U} \bar{X}^{-1/2}\|_F \quad (\text{from (2.27)}) \\ & \leq \|\bar{X}^{-1/2} \bar{U} \bar{V} \bar{X}^{1/2}\|_F \leq \|\bar{X}^{-1/2} \bar{U} \bar{X}^{-1/2}\|_F \|\bar{X}^{1/2} \bar{V} \bar{X}^{1/2}\|_F \\ (2.29) \quad & \leq \alpha(1 - \bar{\theta}) \tau = \alpha \tau_+ \quad (\text{from (2.28)}). \end{aligned}$$

The above inequality implies that

$$\lambda_{\min}((X^+)^{1/2} S^+ (X^+)^{1/2}) \geq (1 - \alpha) \tau_+ > 0.$$

Hence $(X^+)^{1/2} S^+ (X^+)^{1/2} \succ 0$, which gives $S^+ \succ 0$. In view of (2.29), this shows that (2.22) holds for $k + 1$. Finally, (2.25) is an immediate consequence of (2.22). $\square$

3. Global convergence and polynomial complexity. In this section we assume that $\mathcal{F}^*$ is nonempty. Under this assumption we will prove that Algorithm 2.1, with $\epsilon = 0$, is globally convergent in the sense that

$$\lim_{k \rightarrow \infty} \mu_k = 0, \quad \lim_{k \rightarrow \infty} R_d^k = 0, \quad \lim_{k \rightarrow \infty} r_i^k = 0, \quad i = 1, \dots, m.$$

In what follows, we will frequently use the following inequality: for any $M_1, M_2 \in \mathbb{R}^{n \times n}$,

$$(3.1) \quad \|M_1 M_2\|_F \leq \min\{\|M_1\| \|M_2\|_F, \|M_1\|_F \|M_2\|\}$$

(see exercise 20 in section 5.6 of [5]).

LEMMA 3.1. *For any $(\tilde{X}, \tilde{y}, \tilde{S}) \in \mathcal{F}$, we have*

$$\begin{aligned} \psi(S \bullet X^0 + S^0 \bullet X) &= S \bullet X + \psi^2 S^0 \bullet X^0 + (1 - \psi)^2 \tilde{S} \bullet \tilde{X} \\ &\quad + \psi(1 - \psi)(S^0 \bullet \tilde{X} + \tilde{S} \bullet X^0) - (1 - \psi)(S \bullet \tilde{X} + \tilde{S} \bullet X). \end{aligned}$$

Proof. Let

$$\begin{aligned} X' &= X - \psi X^0 - (1 - \psi) \tilde{X}, \\ y' &= y - \psi y^0 - (1 - \psi) \tilde{y}, \\ S' &= S - \psi S^0 - (1 - \psi) \tilde{S}. \end{aligned}$$

From (2.1) and (2.23), it is easily seen that (X', y', S') satisfies

$$\begin{aligned} A_i \bullet X' &= 0, i = 1, \dots, m, \\ \sum_{i=1}^m y'_i A_i + S' &= 0, \end{aligned}$$

which implies $X' \bullet S' = 0$, i.e.,

$$(3.2) \quad [S - \psi S^0 - (1 - \psi)\tilde{S}] \bullet [X - \psi X^0 - (1 - \psi)\tilde{X}] = 0.$$

By expanding (3.2) we obtain the desired result. $\square$

LEMMA 3.2. *Assume that $\mathcal{F}^*$ is nonempty. Then for any $(X^*, y^*, S^*) \in \mathcal{F}^*$ and $(X, y, S) \in \mathcal{N}(\alpha, \tau)$ we have*

$$(3.3a) \quad X \bullet S^0 + X^0 \bullet S \leq (2 + \alpha + \zeta)n\tau_0,$$

$$(3.3b) \quad X \bullet S^* + X^* \bullet S \leq ((1 + \alpha + \psi)/(1 - \psi) + \zeta)n\tau,$$

where

$$(3.4) \quad \zeta = (X^0 \bullet S^* + X^* \bullet S^0)/(X^0 \bullet S^0).$$

Proof. The results follow by using Lemma 3.1 with $(\tilde{X}, \tilde{y}, \tilde{S}) = (X^*, y^*, S^*)$, Theorem 2.6, and the fact that $S^* \bullet X^* = 0$. $\square$

Lemma 3.2 shows that the pair (X^k, S^k) generated by Algorithm 2.1 is bounded. More precisely, we have the following corollary, which can easily be deduced from Lemma 3.2 and Theorem 2.6.

COROLLARY 3.3.

$$\begin{aligned} \|X^{1/2}(S^0)^{1/2}\|_F &\leq \sqrt{(2 + \alpha + \zeta)n\tau_0}, \\ \|S^{1/2}(X^0)^{1/2}\|_F &\leq \sqrt{(2 + \alpha + \zeta)n\tau_0}, \\ \|X^{1/2}\|_F &\leq \|(S^0)^{-1/2}\| \sqrt{(2 + \alpha + \zeta)n\tau_0}, \\ \|S^{1/2}\|_F &\leq \|(X^0)^{-1/2}\| \sqrt{(2 + \alpha + \zeta)n\tau_0}, \\ \|X^{1/2}S^{1/2}\|^2 &= \|X^{1/2}SX^{1/2}\| \leq (1 + \alpha)\tau, \\ \|X^{-1/2}S^{-1/2}\|^2 &= \|X^{-1/2}S^{-1}X^{-1/2}\| \leq \frac{1}{(1 - \alpha)\tau}. \end{aligned}$$

LEMMA 3.4. *Suppose $(\tilde{X}, \tilde{y}, \tilde{S}) \in \mathcal{F}$, and denote*

$$\begin{aligned} T &= \psi[2X^{1/2}(S^0 - \tilde{S})X^{1/2} + X^{-1/2}(X^0 - \tilde{X})SX^{1/2} + X^{1/2}S(X^0 - \tilde{X})X^{-1/2}] \\ &\quad - 2X^{1/2}SX^{1/2}, \\ T_x &= \psi X^{-1/2}(X^0 - \tilde{X})X^{-1/2}, \\ T_s &= \psi X^{1/2}(S^0 - \tilde{S})X^{1/2}. \end{aligned}$$

Then the quantity δ defined by (2.8) satisfies the inequality

$$\delta \leq \frac{1}{\tau^2} \left(\tau \|T_x\|_F + \frac{\|T\|_F}{2(1 - \alpha)} \right) \left(\|T_s\|_F + \frac{\|T\|_F}{2(1 - \alpha)} \right).$$

Proof. It is easily seen that $(U + \psi(X^0 - \tilde{X}), w + \psi(y^0 - \tilde{y}), V + \psi(S^0 - \tilde{S}))$ satisfies (2.18) with $H = T/2$, where (U, w, V) is the predictor direction defined by (2.4). Hence, according to Lemma 2.3, we have

$$\begin{aligned}\tau \|X^{-1/2}(U + \psi(X^0 - \tilde{X}))X^{-1/2}\|_F &\leq \|T/2\|_F/(1 - \alpha), \\ \|X^{1/2}(V + \psi(S^0 - \tilde{S}))X^{1/2}\|_F &\leq \|T/2\|_F/(1 - \alpha).\end{aligned}$$

Therefore,

$$\begin{aligned}\tau \|X^{-1/2}UX^{-1/2}\| &\leq \tau \|T_x\| + \|T/2\|_F/(1 - \alpha), \\ \|X^{1/2}VX^{1/2}\| &\leq \|T_s\| + \|T/2\|_F/(1 - \alpha),\end{aligned}$$

and the lemma follows by virtue of (2.8). $\square$

LEMMA 3.5. *Under the hypothesis of Lemma 3.4 we have*

$$(3.5) \quad \delta \leq \left(\frac{(3 - \alpha)(2 + \alpha + \zeta)n}{(1 - \alpha)^2} d_0 + \frac{1 + \alpha}{1 - \alpha} \sqrt{n} \right)^2,$$

where

$$d_0 = \max(\|(X^0)^{-1/2}(X^0 - \tilde{X})(X^0)^{-1/2}\|_F, \|(S^0)^{-1/2}(S^0 - \tilde{S})(S^0)^{-1/2}\|_F).$$

Proof. Using the notation of Lemma 3.4, and Corollary 3.3, we can write

$$\begin{aligned}\|T\|_F &\leq 2\psi \|X^{1/2}(S^0 - \tilde{S})X^{1/2}\|_F + 2\psi \|X^{-1/2}(X^0 - \tilde{X})SX^{1/2}\|_F + 2\|X^{1/2}SX^{1/2}\|_F \\ &= 2\psi \|X^{1/2}(S^0)^{1/2}(S^0)^{-1/2}(S^0 - \tilde{S})(S^0)^{-1/2}(S^0)^{1/2}X^{1/2}\|_F \\ &\quad + 2\psi \|X^{-1/2}S^{-1/2}S^{1/2}(X^0)^{1/2}(X^0)^{-1/2}(X^0 - \tilde{X})(X^0)^{-1/2}(X^0)^{1/2}S^{1/2}X^{1/2}\|_F \\ &\quad + 2\|X^{1/2}SX^{1/2}\|_F \\ &\leq 2\psi \|X^{1/2}(S^0)^{1/2}\|^2 \|(S^0)^{-1/2}(S^0 - \tilde{S})(S^0)^{-1/2}\|_F \\ &\quad + 2\psi \|X^{-1/2}S^{-1/2}\|^2 \|S^{1/2}(X^0)^{1/2}\|^2 \|X^{1/2}S^{1/2}\| \|(X^0)^{-1/2}(X^0 - \tilde{X})(X^0)^{-1/2}\|_F \\ &\quad + 2\sqrt{n} \|X^{1/2}SX^{1/2}\| \\ &\leq 2\psi(2 + \alpha + \zeta)n\tau_0 d_0 \left(1 + \frac{1 + \alpha}{1 - \alpha}\right) + 2\sqrt{n}(1 + \alpha)\tau \\ &= 2\tau[2(2 + \alpha + \zeta)nd_0/(1 - \alpha) + (1 + \alpha)\sqrt{n}].\end{aligned}$$

Also,

$$\begin{aligned}\|T_x\|_F &\leq \psi \|X^{-1/2}S^{-1/2}S^{1/2}(X^0)^{1/2}(X^0)^{-1/2}(X^0 - \tilde{X})(X^0)^{-1/2}(X^0)^{1/2}S^{1/2}S^{-1/2}X^{-1/2}\|_F \\ &\leq \psi \|X^{-1/2}S^{-1/2}\|^2 \|S^{1/2}(X^0)^{1/2}\|^2 d_0 \\ &\leq \frac{\psi(2 + \alpha + \zeta)n\tau_0}{\tau(1 - \alpha)} d_0 \\ &= \frac{(2 + \alpha + \zeta)n}{1 - \alpha} d_0,\end{aligned}$$

and

$$\begin{aligned}\|T_s\|_F &\leq \psi \|X^{1/2}(S^0)^{1/2}(S^0)^{-1/2}(S^0 - \tilde{S})(S^0)^{-1/2}(S^0)^{1/2}X^{1/2}\| \\ &\leq \psi \|X^{1/2}(S^0)^{1/2}\|^2 d_0 \\ &\leq \psi(2 + \alpha + \zeta)n\tau_0 d_0 = (2 + \alpha + \zeta)n\tau d_0.\end{aligned}$$

Then (3.5) follows from Lemma 3.4. $\square$

According to Lemmas 2.5 and 3.5, it follows that if $\mathcal{F}^*$ is not empty, then the step length $\bar{\theta}_k$ defined by (2.9) is bounded away from 0. This implies global convergence as shown in the following theorem.

THEOREM 3.6. *If $\mathcal{F}^*$ is not empty, then Algorithm 2.1 is globally convergent at a linear rate. Moreover, the iteration sequence (X^k, y^k, S^k) is bounded and every accumulation point of (X^k, y^k, S^k) belongs to $\mathcal{F}^*$ (i.e., is a primal-dual optimal solution of the SDP problem).*

Using Lemma 3.5, we can easily deduce the following result.

THEOREM 3.7. *Suppose that $\mathcal{F}^*$ is nonempty and that the starting point is chosen such that there is a constant κ independent of n satisfying the inequality*

$$(2 + \alpha + \zeta) \max(\|(X^0)^{-1/2}(X^0 - \tilde{X})(X^0)^{-1/2}\|_F, \|(S^0)^{-1/2}(S^0 - \tilde{S})(S^0)^{-1/2}\|_F) \leq \kappa n^{-1/2}.$$

Then Algorithm 2.1 terminates in at most $O(\sqrt{n} \ln(\epsilon_0/\epsilon))$ iterations, where

$$(3.6) \quad \epsilon_0 = \max\{X^0 \bullet S^0, \|R_d^0\|, |r_i^0|, i = 1, \dots, m\}.$$

THEOREM 3.8. *Suppose $X^0 = S^0 = \rho I$, where $\rho > 0$ is a constant such that $\|X^*\| \leq \rho$, $\|S^*\| \leq \rho$ for some $(X^*, y^*, S^*) \in \mathcal{F}^*$. Then the step length $\bar{\theta}_k$ defined by (2.9) satisfies the inequality*

$$\bar{\theta}_k \geq 1/(\omega n),$$

with $\omega \equiv 1 + 29/\sqrt{\beta - \alpha}$.

Proof. According to Lemma 3.2, we have

$$\rho(\text{Tr}(X) + \text{Tr}(S)) \leq (2 + \alpha + \zeta)n\tau_0 = (2 + \alpha + \zeta)n\rho^2,$$

i.e.,

$$\sum_{i=1}^n (\lambda_i(X) + \lambda_i(S)) \leq (2 + \alpha + \zeta)\rho n.$$

In what follows we will frequently use the fact that $\alpha < 0.5$. Since $X^* \bullet S^* = 0$, we get the relation

$$\begin{aligned} \zeta &= (S^* \bullet X^0 + X^* \bullet S^0)/(X^0 \bullet S^0) \\ &= (\text{Tr}(X^*) + \text{Tr}(S^*))/(n\rho) \\ &\leq 1, \end{aligned}$$

which implies

$$(3.7) \quad \|X^{1/2}\|_F^2 + \|S^{1/2}\|_F^2 = \sum_{i=1}^n (\lambda_i(X) + \lambda_i(S)) \leq (3 + \alpha)\rho n.$$

It is easily seen that

$$(3.8) \quad \|X^0 - X^*\| \leq \rho, \quad \|S^0 - S^*\| \leq \rho.$$

Applying (3.7), (3.8), Corollary 3.3, and Lemma 3.4 with $(\tilde{X}, \tilde{y}, \tilde{S}) = (X^*, y^*, S^*)$ we have

$$(3.9) \quad \|X^{1/2}(S^0 - S^*)X^{1/2}\| \leq \|X^{1/2}\|_F^2 \|S^0 - S^*\| \leq 3.5\rho^2 n,$$

$$(3.10) \quad \begin{aligned} & \|X^{-1/2}(X^0 - X^*)SX^{1/2}\| \\ &= \|(X^{-1/2}S^{-1/2})S^{1/2}(X^0 - X^*)S^{1/2}(S^{1/2}X^{1/2})\| \\ &\leq \|X^{-1/2}S^{-1/2}\| \|S^{1/2}X^{1/2}\| \|S^{1/2}\|_F^2 \|X^0 - X^*\| \\ &\leq 6.1\rho^2 n. \end{aligned}$$

In view of (3.9), (3.10), and Lemma 3.4, we obtain

$$(3.11) \quad \begin{aligned} \|T\|_F &\leq 2[\psi\|X^{1/2}(S^0 - S^*)X^{1/2}\| + \psi\|X^{-1/2}(X^0 - X^*)SX^{1/2}\| \\ &\quad + \|X^{1/2}SX^{1/2}\|] \\ &< 2\tau[9.6n + 1.5] \leq 23\tau n, \end{aligned}$$

$$(3.12) \quad \begin{aligned} \tau\|T_x\|_F &= \tau\psi\|X^{-1/2}(X^0 - X^*)X^{-1/2}\| \\ &= \tau\psi\|(X^{-1/2}S^{-1/2})S^{1/2}(X^0 - X^*)S^{1/2}(S^{-1/2}X^{-1/2})\| \\ &\leq \tau\psi\|S^{1/2}\|_F^2 \|X^{-1/2}S^{-1/2}\|^2 \|X^0 - X^*\| \\ &< 7\tau n, \end{aligned}$$

$$(3.13) \quad \begin{aligned} \|T_s\|_F &= \psi\|X^{1/2}(S^0 - S^*)X^{1/2}\| \\ &\leq \psi\|X^{1/2}\|_F^2 \|S^0 - S^*\| \\ &\leq 3.5\tau n. \end{aligned}$$

Therefore, from Lemma 3.4, (3.11), (3.12), and (3.13), we obtain

$$\begin{aligned} \delta &\leq \frac{1}{\tau^2} \left(\tau\|T_x\|_F + \frac{\|T\|_F}{2(1-\alpha)} \right) \left(\|T_s\|_F + \frac{\|T\|_F}{2(1-\alpha)} \right) \\ &\leq 795n^2. \end{aligned}$$

Consequently, $\bar{\theta} > 1/(\omega n)$. $\square$

In the following corollary we summarize the complexity results for standard starting points of the form $X^0 = S^0 = \rho I$.

COROLLARY 3.9. *Assume that in Algorithm 2.1 we choose a starting point of the form $X^0 = S^0 = \rho I$, where $\rho > 0$ is a constant. Let ϵ_0 be given by (3.6) and let $\epsilon > 0$ be arbitrary. Then the following statements hold:*

- (i) *If $\mathcal{F}^* \neq \emptyset$, then the algorithm terminates with an ϵ -approximate solution $(X^k, y^k, S^k) \in \mathcal{F}_\epsilon$ in a finite number of steps $k = K_\epsilon < \infty$.*
- (ii) *If $\rho \geq \max\{\|X^*\|, \|S^*\|\}$ for some $(X^*, y^*, S^*) \in \mathcal{F}^*$, then*

$$K_\epsilon = O(n \ln(\epsilon_0/\epsilon)).$$

- (iii) *For any choice of $\rho > 0$ there is an index $k = \hat{K}_\epsilon = O(n \ln(\epsilon_0/\epsilon))$ such that either*
 - (iiia) $(X^k, y^k, S^k) \in \mathcal{F}_\epsilon$
 - or
 - (iiib) $\bar{\theta} \leq 1/(\omega n)$,*and in the latter case there is no solution $(X^*, y^*, S^*) \in \mathcal{F}^*$ with $\rho \geq \max\{\|X^*\|, \|S^*\|\}$.*

4. Superlinear convergence. The next two lemmas are well known and can be easily proved.

LEMMA 4.1. *Let $A_1, A_2 \in S_+^p$. Then $A_1 \bullet A_2 = 0$ if and only if $A_1 A_2 = A_2 A_1 = 0$.*

LEMMA 4.2. *Let $(X^*, y^*, S^*) \in \mathcal{F}^*$. Then there exists an orthogonal matrix $Q = (q_1, q_2, \dots, q_n)$ such that $Q^T X^* Q$ and $Q^T S^* Q$ are diagonal matrices. In other words, $q_1, q_2, \dots, q_n$ are eigenvectors of X^* and S^* .*

DEFINITION 4.3. *A triple $(X^*, y^*, S^*) \in \mathcal{F}^*$ is called a strictly complementary solution of (1.3) if $X^* + S^* \succ 0$.*

In this section we investigate the asymptotic behavior of Algorithm 2.1. We will propose a sufficient condition for the superlinear convergence of Algorithm 2.1.

Assumption 1. The SDP problem has a strictly complementary solution.

Let (X^*, y^*, S^*) be a strictly complementary solution of (1.3) and $Q = (q_1, \dots, q_n)$ be an orthogonal matrix such that $q_1, \dots, q_n$ are eigenvectors of X^* and S^* . Define

$$\mathbb{B} = \{i : q_i^T X^* q_i > 0\}, \quad \mathbb{N} = \{i : q_i^T S^* q_i > 0\}.$$

It is easily seen that $\mathbb{B} \cup \mathbb{N} = \{1, 2, \dots, n\}$. For simplicity, let us assume without loss of generality that

$$Q^T X^* Q = \begin{pmatrix} \Lambda_B & 0 \\ 0 & 0 \end{pmatrix}, \quad Q^T S^* Q = \begin{pmatrix} 0 & 0 \\ 0 & \Lambda_N \end{pmatrix},$$

where Λ_B and Λ_N are diagonal matrices. Here and in what follows, if we write a matrix M in the block form

$$M = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix},$$

then we assume that the dimensions of M_{11} and M_{22} are $|\mathbb{B}| \times |\mathbb{B}|$ and $|\mathbb{N}| \times |\mathbb{N}|$, respectively. In the next lemma we use the following notations:

$$\begin{aligned} Q^T X^{1/2} Q &= (x_1, x_2, \dots, x_n), \\ Q^T S^{1/2} Q &= (s_1, s_2, \dots, s_n), \\ Q^T X^{-1/2} Q &= (\hat{x}_1, \hat{x}_2, \dots, \hat{x}_n), \\ Q^T S^{-1/2} Q &= (\hat{s}_1, \hat{s}_2, \dots, \hat{s}_n), \end{aligned}$$

where $x_i, s_i, \hat{x}_i, \hat{s}_i \in \mathbb{R}^n$, $i = 1, \dots, n$.

LEMMA 4.4. *Under Assumption 1,*

$$\begin{aligned} \|x_i\| &= O(\sqrt{\tau}), \quad \|s_i\| = O(1) \quad \forall i \in \mathbb{N}; \\ \|x_i\| &= O(1), \quad \|s_i\| = O(\sqrt{\tau}) \quad \forall i \in \mathbb{B}; \\ \|\hat{x}_i\| &= O(1), \quad \|\hat{s}_i\| = O(1/\sqrt{\tau}) \quad \forall i \in \mathbb{B}; \\ \|\hat{x}_i\| &= O(1/\sqrt{\tau}), \quad \|\hat{s}_i\| = O(1) \quad \forall i \in \mathbb{N}. \end{aligned}$$

Proof. Because the sequence $\{(X^k, S^k)\}$ is bounded, we have

$$\|x_i\| = O(1), \quad \|s_i\| = O(1) \quad \forall i \in \{1, \dots, n\}.$$

In view of (3.3b), we get

$$\text{Tr}(X^{1/2} S^* X^{1/2}) + \text{Tr}(S^{1/2} X^* S^{1/2}) \leq ((1 + \alpha + \psi)/(1 - \psi) + \zeta)n\tau \leq \nu_1 \tau,$$

where

$$\nu_1 = ((2 + \alpha)/(1 - \psi_1) + \zeta)n.$$

Let $\nu = \lambda_{\min}(X^* + S^*)$. Since

$$\begin{aligned} \text{Tr}(X^{1/2}S^*X^{1/2}) &= \text{Tr}(Q^T X^{1/2} Q Q^T S^* Q Q^T X^{1/2} Q) \\ &= \text{Tr}((x_1, \dots, x_n) Q^T S^* Q (x_1, \dots, x_n)^T) \\ &= \sum_{i \in \mathbb{N}} (q_i^T S^* q_i) \|x_i\|^2 \end{aligned}$$

and $q_i^T S^* q_i \geq \nu > 0$ for each $i \in \mathbb{N}$, we have

$$\|x_i\|^2 \leq \nu_2 \tau \quad \forall i \in \mathbb{N},$$

where $\nu_2 = \nu_1/\nu$. Hence $\|x_i\| = O(\sqrt{\tau}) \quad \forall i \in \mathbb{N}$. Similarly, $\|s_i\| = O(\sqrt{\tau}) \quad \forall i \in \mathbb{B}$. From Theorem 2.6, we obtain

$$X^{1/2} S X^{1/2} - (1 - \alpha)\tau I \succeq 0,$$

which implies

$$S - (1 - \alpha)\tau X^{-1} \succeq 0,$$

i.e.,

$$[Q^T S^{1/2} Q]^2 - (1 - \alpha)\tau [Q^T X^{-1/2} Q]^2 \succeq 0.$$

Therefore,

$$e_i^T [Q^T S^{1/2} Q]^2 e_i \geq (1 - \alpha)\tau e_i^T [Q^T X^{-1/2} Q]^2 e_i, \quad 1 \leq i \leq n.$$

Hence, for any $i \in \mathbb{B}$ we can write

$$\begin{aligned} \|\hat{x}_i\|^2 &= e_i^T [Q^T X^{-1/2} Q]^2 e_i \\ &\leq \frac{e_i^T [Q^T S^{1/2} Q]^2 e_i}{(1 - \alpha)\tau} \\ &= \frac{\|s_i\|^2}{(1 - \alpha)\tau} \\ &\leq \nu_2 / (1 - \alpha). \end{aligned}$$

Also, for any $i \in \mathbb{N}$,

$$\begin{aligned} \|\hat{x}_i\|^2 &= e_i^T [Q^T X^{-1/2} Q]^2 e_i \\ &\leq \frac{\|s_i\|^2}{(1 - \alpha)\tau} = O(1/\tau). \end{aligned}$$

Therefore, $\|\hat{x}_i\| = O(1) \quad \forall i \in \mathbb{B}$, $\|\hat{x}_i\| = O(1/\sqrt{\tau}) \quad \forall i \in \mathbb{N}$. Similarly, by considering $X - (1 - \alpha)\tau S^{-1} \succeq 0$, we can show that $\|\hat{s}_i\| = O(1) \quad \forall i \in \mathbb{N}$ and $\|\hat{x}_i\| = O(1/\sqrt{\tau}) \quad \forall i \in \mathbb{B}$. $\square$

Using Lemma 4.4, we can write

$$(4.1) \quad Q^T X^{1/2} Q = \begin{pmatrix} O(1) & O(\sqrt{\tau}) \\ O(\sqrt{\tau}) & O(\sqrt{\tau}) \end{pmatrix}, \quad Q^T X^{-1/2} Q = \begin{pmatrix} O(1) & O(1) \\ O(1) & O(1/\sqrt{\tau}) \end{pmatrix},$$

$$(4.2) \quad Q^T S^{1/2} Q = \begin{pmatrix} O(\sqrt{\tau}) & O(\sqrt{\tau}) \\ O(\sqrt{\tau}) & O(1) \end{pmatrix}, \quad Q^T S^{-1/2} Q = \begin{pmatrix} O(1/\sqrt{\tau}) & O(1) \\ O(1) & O(1) \end{pmatrix}.$$

Let us define a linear manifold:

$$(4.3) \quad \mathcal{M} \equiv \left\{ (X', y', S') \in \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n : A_i \bullet X' = b_i, \quad i = 1, \dots, m, \right. \\ \left. \sum_{i=1}^m y'_i A_i + S' = C, \right. \\ \left. q_i^T X' q_j = 0 \text{ if } i \text{ or } j \in \mathbb{N}, \right. \\ \left. q_i^T S' q_j = 0 \text{ if } i \text{ or } j \in \mathbb{B} \right\}.$$

It is easily seen that if $(X', y', S') \in \mathcal{M}$, then

$$Q^T X' Q = \begin{pmatrix} M_B & 0 \\ 0 & 0 \end{pmatrix}, \quad Q^T S' Q = \begin{pmatrix} 0 & 0 \\ 0 & M_N \end{pmatrix}.$$

LEMMA 4.5. *Under Assumption 1, $\mathcal{F}^* \subset \mathcal{M}$.*

Proof. For any $(\bar{X}^*, \bar{y}^*, \bar{S}^*) \in \mathcal{F}^*$, we have

$$A_i \bullet (\bar{X}^* - X^*) = 0, \quad i = 1, \dots, m, \quad \sum_{i=1}^m A_i (\bar{y}_i^* - y_i^*) + (\bar{S}^* - S^*) = 0.$$

Hence,

$$(\bar{S}^* - S^*) \bullet (\bar{X}^* - X^*) = 0,$$

which implies

$$\bar{S}^* \bullet X^* + S^* \bullet \bar{X}^* = \bar{S}^* \bullet \bar{X}^* + S^* \bullet X^* = 0.$$

Then, $\bar{S}^* \bullet X^* = S^* \bullet \bar{X}^* = 0$. By Lemma 4.1, we have

$$\bar{S}^* X^* = X^* \bar{S}^* = 0,$$

which implies

$$[Q^T \bar{S}^* Q][Q^T X^* Q] = [Q^T X^* Q][Q^T \bar{S}^* Q] = 0,$$

i.e.,

$$(4.4) \quad [q_i^T \bar{S}^* q_j][q_j^T X^* q_j] = [q_i^T X^* q_i][q_i^T \bar{S}^* q_j] = 0.$$

If i or $j \in \mathbb{B}$, then $q_j^T X^* q_j$ or $q_i^T X^* q_i$ is positive, which implies $q_i^T \bar{S}^* q_j = 0$ according to (4.4). Similarly, we can show that $q_i^T \bar{X}^* q_j = 0$ whenever i or $j \in \mathbb{N}$. Therefore, $(\bar{X}^*, \bar{y}^*, \bar{S}^*) \in \mathcal{M}$, which gives $\mathcal{F}^* \subset \mathcal{M}$. $\square$

LEMMA 4.6. *Under Assumption 1, every accumulation point of (X^k, y^k, S^k) is a strictly complementary solution of (1.3).*

Proof. Suppose $(X'', y'', S'') \in \mathcal{F}^*$ is an accumulation point. Let us assume, without loss of generality, that $(X^k, y^k, S^k) \rightarrow (X'', y'', S'')$. Then, according to Lemma 4.5,

$$\begin{aligned} Q^T X^k Q &\rightarrow Q^T X'' Q = \begin{pmatrix} M_B & 0 \\ 0 & 0 \end{pmatrix}, \\ Q^T S^k Q &\rightarrow Q^T S'' Q = \begin{pmatrix} 0 & 0 \\ 0 & M_N \end{pmatrix} \end{aligned}$$

for some symmetric positive semidefinite matrices M_B and M_N . In order to show $X'' + S'' \succ 0$, it remains to prove that M_B and M_N are nonsingular and therefore positive definite. From Lemma 4.4 or (4.1), we have

$$\begin{aligned} Q^T (X^k)^{1/2} Q &= \begin{pmatrix} M_1^k & O(\sqrt{\tau_k}) \\ O(\sqrt{\tau_k}) & O(\sqrt{\tau_k}) \end{pmatrix}, \\ Q^T (X^k)^{-1/2} Q &= \begin{pmatrix} M_2^k & O(1) \\ O(1) & O(1/\sqrt{\tau_k}) \end{pmatrix}, \end{aligned}$$

where $M_1^k = O(1)$, $M_2^k = O(1)$. Since $((X^k)^{1/2}, (S^k)^{1/2})$ is bounded, it has an accumulation point. Without loss of generality, we may assume

$$Q^T (X^k)^{1/2} Q \rightarrow \begin{pmatrix} \widetilde{M}_1 & 0 \\ 0 & 0 \end{pmatrix}.$$

Then, $M_1^k \rightarrow \widetilde{M}_1$. Since

$$[Q^T (X^k)^{1/2} Q][Q^T (X^k)^{-1/2} Q] = I,$$

we obtain

$$M_1^k M_2^k = I_B - O(\sqrt{\tau_k}) \rightarrow I_B,$$

which implies

$$(M_1^k)^{-1} = M_2^k (I_B - O(\sqrt{\tau_k}))^{-1} = O(1).$$

Hence $\widetilde{M}_1$ must be nonsingular. Obviously $\widetilde{M}_1^2 = M_B$, so that M_B is nonsingular. Similarly, we can show that M_N is nonsingular. $\square$

In the next theorem, we propose a sufficient condition for the superlinear convergence of Algorithm 2.1. Let us define

$$(4.5) \quad \eta_k = \eta_k(\Gamma) = \frac{1}{\tau_k} \|(X^k)^{-1/2} (X^k - \check{X}^k)(S^k - \check{S}^k)(X^k)^{1/2}\|_F,$$

where $(\check{X}^k, \check{y}^k, \check{S}^k)$ is the solution of the following minimization problem:

$$(4.6) \quad \min\{\|(X^k)^{-1/2} (X^k - X')(S^k - S')(X^k)^{1/2}\|_F : (X', y', S') \in \mathcal{M}, \|(X', S')\|_F \leq \Gamma\},$$

and Γ is a constant such that $\|(X^k, S^k)\|_F \leq \Gamma \forall k$. For a definition of $\|(X, S)\|_F$ see the notation introduced at the end of section 1. Note that every accumulation point of (X^k, y^k, S^k) belongs to the feasible set of the above minimization problem and the feasible set is bounded. Therefore $(\check{X}^k, \check{S}^k)$ exists for each k .

Assumption 2. $\eta_k \rightarrow 0$ as $k \rightarrow \infty$.

THEOREM 4.7. *Under Assumptions 1 and 2, Algorithm 2.1 is superlinearly convergent. Moreover, if there exists a constant $\sigma > 0$ such that $\eta_k = O(\tau_k^\sigma)$, then the convergence has Q -order of at least $1 + \sigma$ in the sense that $\mu_{k+1} = O(\mu_k^{1+\sigma})$.*

Proof. By Lemma 2.5, it remains to prove that $\delta_k \rightarrow 0$ as $k \rightarrow \infty$. For simplicity, let us omit the index k . It is easily seen that $(U + X - \check{X}, w + y - \check{y}, V + S - \check{S})$ satisfies (2.18) with

$$\begin{aligned} 2H &= X^{-1/2}[X(S - \check{S}) + (X - \check{X})S]X^{1/2} \\ &\quad + X^{1/2}[(S - \check{S})X + S(X - \check{X})]X^{-1/2} - 2X^{1/2}SX^{1/2} \\ &= \Delta + \Delta^T, \end{aligned}$$

where

$$(4.7) \quad \Delta = X^{-1/2}(X - \check{X})(S - \check{S})X^{1/2}.$$

Here we have used the relation $\check{X}\check{S} = \check{S}\check{X} = 0$. Then,

$$(4.8) \quad \|H\|_F \leq \|\Delta\|_F = \eta\tau.$$

Denoting

$$\begin{aligned} \Delta_x &= X^{-1/2}(U + X - \check{X})X^{-1/2}, \\ \Delta_s &= X^{1/2}(V + S - \check{S})X^{1/2}, \end{aligned}$$

and applying Lemma 2.3, we obtain

$$\tau\|\Delta_x\|_F \leq \|H\|_F/(1 - \alpha) \leq \eta\tau/(1 - \alpha),$$

which implies

$$(4.9) \quad \|\Delta_x\|_F \leq \eta/(1 - \alpha).$$

Similarly,

$$(4.10) \quad \|\Delta_s\|_F \leq \|H\|_F/(1 - \alpha) \leq \eta\tau/(1 - \alpha).$$

By Lemma 4.4 and the fact that $(\check{X}, \check{y}, \check{S}) \in \mathcal{M}$, we have

$$\begin{aligned} \|X^{-1/2}(X - \check{X})X^{-1/2}\|_F &= \|I - X^{-1/2}\check{X}X^{-1/2}\|_F \\ &\leq \|I\|_F + \|X^{-1/2}\check{X}X^{-1/2}\|_F \\ &= \sqrt{n} + \|Q^T X^{-1/2} Q Q^T \check{X} Q Q^T X^{-1/2} Q\|_F \\ &= \sqrt{n} + \|(\widehat{x}_1, \dots, \widehat{x}_n) Q^T \check{X} Q (\widehat{x}_1, \dots, \widehat{x}_n)^T\|_F \\ &= \sqrt{n} + \|\sum_{i,j \in \mathbb{B}} (q_i^T \check{X} q_j) \widehat{x}_i \widehat{x}_j^T\|_F \\ (4.11) \quad &= O(1). \end{aligned}$$

Similarly,

$$(4.12) \quad \|S^{-1/2}(S - \check{S})S^{-1/2}\|_F = O(1).$$

Let us observe that

$$\begin{aligned}
X^{-1/2}UVX^{1/2} &= (X^{-1/2}UX^{-1/2})(X^{1/2}VX^{1/2}) \\
&= (\Delta_x - X^{-1/2}(X - \check{X})X^{-1/2})(\Delta_s - X^{1/2}(S - \check{S})X^{1/2}) \\
&= \Delta_x\Delta_s - (X^{-1/2}(X - \check{X})X^{-1/2})\Delta_s \\
&\quad - \Delta_x X^{1/2}S^{1/2}(S^{-1/2}(S - \check{S})S^{-1/2})S^{1/2}X^{1/2} + \Delta.
\end{aligned}$$

Then from (4.8), (4.9), (4.10), (4.11), (4.12), and Corollary 3.3, we have

$$\begin{aligned}
\|X^{-1/2}UVX^{1/2}\|_F &\leq \|\Delta_x\|_F\|\Delta_s\|_F + \|X^{-1/2}(X - \check{X})X^{-1/2}\|_F\|\Delta_s\|_F \\
&\quad + \|X^{1/2}S^{1/2}\|^2\|\Delta_x\|_F\|S^{-1/2}(S - \check{S})S^{-1/2}\|_F + \|\Delta\|_F \\
&= O(\eta\tau).
\end{aligned}$$

Hence, $\delta_k = O(\eta_k) \rightarrow 0$. Finally, if $\eta_k = O(\tau_k^\sigma)$ for some constant $\sigma > 0$, then we have $\delta_k = O(\tau_k^\sigma)$. From Lemma 2.5,

$$\begin{aligned}
1 - \bar{\theta} &\leq 1 - \frac{2}{1 + \sqrt{1 + 4\delta/(\beta - \alpha)}} \\
&= \frac{\sqrt{1 + 4\delta/(\beta - \alpha)} - 1}{\sqrt{1 + 4\delta/(\beta - \alpha)} + 1} \\
&= \frac{4\delta/(\beta - \alpha)}{(\sqrt{1 + 4\delta/(\beta - \alpha)} + 1)^2} \\
&\leq \delta/(\beta - \alpha) = O(\tau^\sigma).
\end{aligned}$$

Therefore, $\tau_{k+1} = (1 - \bar{\theta}_k)\tau_k = O(\tau_k^{1+\sigma})$. Recall (2.25). We then obtain $\mu_{k+1} = O(\mu_k^{1+\sigma})$. $\square$

We mention that our sufficient condition in the above theorem is satisfied under conditions (A), (B), and (C) of Kojima, Shida, and Shindoh [8].

We end this paper by giving two special cases of SDP for which Algorithm 2.1 is quadratically convergent.

PROPOSITION 4.8. *If $\mathbb{B} = \{1, \dots, n\}$ or $\mathbb{N} = \{1, \dots, n\}$, then Algorithm 2.1 is quadratically convergent.*

Proof. Let $(\tilde{X}, \tilde{y}, \tilde{S})$ be the solution of the minimization problem

$$(4.13) \quad \min_{(X', y', S') \in \mathcal{M}} \|(X - X', S - S')\|_F.$$

We will show that

$$(4.14) \quad \|(X - \check{X}, S - \check{S})\|_F = O(\tau).$$

We will prove (4.14) only for $\mathbb{B} = \{1, \dots, n\}$, and a similar proof applies to the case $\mathbb{N} = \{1, \dots, n\}$. The manifold $\mathcal{M}$ reduces to

$$\begin{aligned}
(4.15) \quad &\left\{ (X', y', S') \in \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n : A_i \bullet X' = b_i, \quad i = 1, \dots, m, \right. \\
&\quad \left. \sum_{i=1}^m y'_i A_i + S' = C, \quad S' = 0 \right\}.
\end{aligned}$$

Assume by contradiction that (4.14) does not hold, i.e., there exists a subsequence such that

$$(4.16) \quad \xi_k = \|(X^k - \check{X}^k, S^k - \check{S}^k)\|_F / \tau_k \rightarrow \infty \text{ as } k \rightarrow \infty.$$

Let us define

$$(4.17) \quad (\Delta X^k, \Delta y^k, \Delta S^k) = \frac{1}{\tau_k \xi_k} (X^k - \check{X}^k, y^k - \check{y}^k, S^k - \check{S}^k).$$

It is easily seen that

$$(4.18a) \quad A_i \bullet \Delta X^k = \frac{1}{\tau_0 \xi_k} A_i \bullet (X^0 - X^*), \quad i = 1, \dots, m,$$

$$(4.18b) \quad \sum_{i=1}^m (\Delta y_i^k) A_i + \Delta S^k = \frac{1}{\tau_0 \xi_k} \left[\sum_{i=1}^m (y_i^0 - y_i^*) A_i + S^0 - S^* \right].$$

Since $\|(\Delta X^k, \Delta S^k)\|_F = 1$ and Δy^k depends linearly on ΔS^k (cf. (4.18)), we deduce that there exists a convergent subsequence of $(\Delta X^k, \Delta y^k, \Delta S^k)$. Without loss of generality we can write

$$(\Delta X^k, \Delta y^k, \Delta S^k) \rightarrow (\Delta X', \Delta y', \Delta S').$$

Letting $k \rightarrow \infty$ in (4.18), we obtain

$$A_i \bullet \Delta X' = 0, \quad i = 1, \dots, m, \quad \sum_{i=1}^m (\Delta y_i') A_i + \Delta S' = 0.$$

From Lemma 4.4, we have for each $i \in \mathbb{B} = \{1, \dots, n\}$,

$$q_i^T (\Delta S^k) q_i = q_i^T S^k q_i / (\xi_k \tau_k) \leq \left(\sum_{i=1}^n \|s_i\|^2 \right) / (\xi_k \tau_k) = O(\tau_k) / (\xi_k \tau_k) = O(1/\xi_k) = o(1),$$

which implies that $\Delta S' = 0$. Hence,

$$(\check{X}^k, \check{y}^k, \check{S}^k) + \nu(\Delta X', \Delta y', \Delta S') \in \mathcal{M} \quad \forall \nu \in \mathbb{R}.$$

Since

$$\begin{aligned} & \frac{\|(X^k - (\check{X}^k + \tau_k \xi_k \Delta X'), S^k - (\check{S}^k + \tau_k \xi_k \Delta S'))\|_F}{\|(X^k - \check{X}^k, S^k - \check{S}^k)\|_F} \\ &= \frac{1}{\tau_k \xi_k} \|(X^k - (\check{X}^k + \tau_k \xi_k \Delta X'), S^k - (\check{S}^k + \tau_k \xi_k \Delta S'))\|_F \\ &= \|(\Delta X^k - \Delta X', \Delta S^k - \Delta S')\|_F \\ &\rightarrow 0 \quad \text{as } k \rightarrow \infty, \end{aligned}$$

$(\check{X}^k, \check{y}^k, \check{S}^k)$ is not the solution of the minimization problem (4.13) for sufficiently large k , which is a contradiction.

It is easily seen from (4.14) that $(\check{X}^k, \check{S}^k)$ is bounded. Hence we can choose Γ in (4.6) such that

$$\max\{\|(X^k, S^k)\|_F, \|(\check{X}^k, \check{S}^k)\|_F\} \leq \Gamma \quad \forall k.$$

Therefore,

$$\begin{aligned}
 \eta &= \frac{1}{\tau} \|X^{-1/2}(X - \check{X})(S - \check{S})X^{1/2}\|_F \\
 &\leq \frac{1}{\tau} \|X^{-1/2}(X - \check{X})(S - \check{S})X^{1/2}\|_F \\
 (4.19) \quad &\leq \frac{1}{\tau} \|X^{-1/2}\| \|X^{1/2}\| \|X - \check{X}\|_F \|S - \check{S}\|_F.
 \end{aligned}$$

From Lemma 4.4 we deduce that if $\mathbb{B} = \{1, \dots, n\}$ or $\mathbb{N} = \{1, \dots, n\}$, then

$$(4.20) \quad \|X^{-1/2}\| \|X^{1/2}\| = O(1).$$

In view of (4.14), (4.19), and (4.20), we have $\eta = O(\tau)$, and the result follows from Theorem 4.7. $\square$

Acknowledgments. The authors would like to thank Professor Masakazu Kojima, Masayuki Shida, and Susumu Shindoh for sending us their paper and kindly pointing out some errors in the first version of the paper. We also thank two anonymous referees for their helpful comments.

REFERENCES

- [1] F. ALIZADEH, *Interior point methods in semidefinite programming with applications to combinatorial optimization*, SIAM J. Optim., 5 (1995), pp. 13–51.
- [2] F. ALIZADEH, J.-P. A. HAEBERLY, AND M. L. OVERTON, *Primal-dual interior point methods for semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 746–768.
- [3] F. ALIZADEH, J.-P. A. HAEBERLY, AND M. L. OVERTON, *Complementarity and nondegeneracy in semidefinite programming*, Math. Programming, 77 (1997), pp. 111–128.
- [4] C. HELMBERG, F. RENDL, R. J. VANDERBEI, AND H. WOLKOWICZ, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.
- [5] R. A. HORN AND C. R. JOHNSON, *Matrix Analysis*, Cambridge University Press, New York, 1985.
- [6] J. JI, F. A. POTRA, AND S. HUANG, *A predictor-corrector method for linear complementarity problems with polynomial complexity and superlinear convergence*, J. Optim. Theory Appl., 84 (1995), pp. 187–199.
- [7] M. KOJIMA, S. KOJIMA, AND S. HARA, *Linear Algebra for Semidefinite Programming*, Research Reports on Information Sciences B-290, Department of Information Sciences, Tokyo Institute of Technology, 2-12-1 Oh-Okayama, Meguro-ku, Tokyo 152, Japan, 1994.
- [8] M. KOJIMA, M. SHIDA, AND S. SHINDOH, *Global and Local Convergence of Predictor-Corrector Infeasible-Interior-Point Algorithms for Semidefinite Programs*, Research Reports on Information Sciences B-305, Department of Information Sciences, Tokyo Institute of Technology, 2-12-1 Oh-Okayama, Meguro-ku, Tokyo 152, Japan, 1995.
- [9] M. KOJIMA, S. SHINDOH, AND S. HARA, *Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.
- [10] S. MIZUNO, F. JARRE, AND J. STOER, *A unified approach to infeasible-interior-point algorithms via geometrical linear complementarity problems*, Appl. Math. Optim., 33 (1996), pp. 315–341.
- [11] S. MIZUNO, M. J. TODD, AND Y. YE, *On adaptive-step primal-dual interior-point algorithms for linear programming*, Math. Oper. Res., 18 (1993), pp. 964–981.
- [12] R. D. C. MONTEIRO, *Primal-dual path-following algorithms for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 663–678.
- [13] Y. E. NESTEROV AND M. J. TODD, *Primal-dual interior-point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.

- [14] M. L. OVERTON, *private communication*.
- [15] R. SHENG AND F. A. POTRA, *A quadratically convergent infeasible-interior-point algorithm for LCP with polynomial complexity*, SIAM J. Optim., 7 (1997), pp. 304–317.
- [16] L. VANDENBERGHE AND S. BOYD, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.
- [17] Y. ZHANG, *On extending some primal-dual interior-point algorithms from linear programming to semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 365–386.

DUALITY IN REVERSE CONVEX OPTIMIZATION*

B. LEMAIRE[†]

Abstract. A duality theorem for the general problem of minimizing an extended real-valued convex function on a locally convex linear space under a reverse convex constraint is considered. In the particular case of the distance to a reverse convex subset in a normed linear space, we recover as a corollary a duality theorem due to C. Franchetti and I. Singer [*Boll. Un. Mat. Ital. B* (5), 17 (1980), pp. 33–43] similar to the one known for the distance to a convex subset. The general theorem also contains the duality principle of Toland–Singer for D.C. optimization.

Key words. duality, D.C. optimization, reverse convex

AMS subject classifications. 90C48, 49N15, 90C26

PII. S1052623495295857

1. Introduction. Reverse convex optimization, that is, minimizing an extended real-valued convex function subject to a reverse convex constraint, constitutes a general framework for a large class of nonconvex optimization problems including D.C. optimization (minimizing or maximizing a difference of two extended real-valued convex functions), maximizing a convex function over a convex set, and minimizing a convex function over a reverse convex set, i.e., the complement of a convex subset of a convex set [10]. This subject has received increased attention in recent years mainly for numerical purposes [10, 14, 26] or from the point of view of necessary and sufficient conditions for optimality with the use of subdifferential calculus of convex analysis and regularizing techniques [8, 9].

Duality theory for nonconvex problems with special structure has yet to be explored, particularly in D.C. optimization [17, 18, 25, 6, 7] and in reverse convex optimization [5, 19, 20, 21, 22, 23, 24]. In the present work, our main purpose is to present a general duality result of the following type: the optimal value of the (primal) problem is equal to the optimal value of a suitable dual problem involving the Legendre–Fenchel conjugates of the data. The approach we adopt here is quite general, and convex analysis theory plays a crucial role in the proof of this nonconvex duality result.

The paper is organized as follows. In section 2 we state the general problem and consider two instances: distance to a reverse convex set and D.C. optimization. Section 3 is devoted to duality results for the two previous examples. In the case of D.C. optimization, this is nothing but the well-known principle of Toland–Singer [25, 17]. The main result is presented in section 4 with a corollary applying to the problem of minimizing a finite-valued convex function subject to a reverse convex constraint. In section 5 we show how the results of section 3 can be obtained as consequences of the general result of section 4.

We use the following extensions of the addition and the product in $\overline{\mathbb{R}}$:

$$\begin{aligned} (+\infty) + (-\infty) &= (-\infty) + (+\infty) = +\infty, & a - b &= a + (-b), \\ 0(-\infty) &= 0, & 0(+\infty) &= +\infty, \end{aligned}$$

*Received by the editors December 6, 1995; accepted for publication (in revised form) March 25, 1997; published electronically September 23, 1998.

<http://www.siam.org/journals/siopt/8-4/29585.html>

[†]Institut de Mathématiques, Université Montpellier II, Place Eugene Bataillon, 34095 Montpellier Cedex 05, France (lemaire@math.univ-montp2.fr).

and refer to [12] for the related calculus rules. For $C \subset X$, δ_C will denote the indicator function of C :

$$\delta_C(x) := \begin{cases} 0 & \text{if } x \in C, \\ +\infty & \text{if } x \notin C. \end{cases}$$

2. General problem and examples. Let X be a Hausdorff locally convex real linear space. We denote by Y its topological dual and by $\langle \cdot, \cdot \rangle$ the canonical pairing between X and Y . Let g, h be two extended real-valued convex functions on X . We are interested in finding (primal problem)

$$\alpha := \inf\{g(x); h(x) > 0, x \in X\}$$

or, in short,

$$\alpha := \inf_{h>0} g,$$

that is, minimizing a convex function under a reverse convex constraint. The unusual setting of the constraint deserves the following comments.

Remark 2.1. 1. Concerning the reverse convex constraint $h > 0$ which defines the complement of a (closed if h is lower semicontinuous) convex subset, it should be noticed that

(i) h may take the values $+\infty$ and $-\infty$,

(ii) as will be shown later, *strict* inequality is crucial for deriving the main result.

Actually, if g and h are both finite valued and continuous, and if the subsets $\{h < 0\}$ and $\{h > 0\}$ are both nonempty, then $\{h \geq 0\}$ is the closure of $\{h > 0\}$ and

$$\inf_{h \geq 0} g = \inf_{h > 0} g.$$

However, in general, the first infimum value above may be strictly smaller than the second one. In the following we shall see two important examples, where g and h are not necessarily continuous; in the first one the two infima are different; in the second one they are equal.

2. Due to a lack of closedness, the infimum value α may not be achieved on the feasible set. However, the present work is not concerned with (primal or dual) optimal solutions. For that, see [5, 2] in the particular case presented as Example 2.1 below.

Example 2.1 (distance to a reverse convex set). Let $(X, \|\cdot\|)$ be a real normed linear space, C a nonempty closed convex subset in X , and a a given point in X .

We are interested in finding

$$\alpha := d(a, X \setminus C) := \inf_{x \notin C} \|x - a\|.$$

This is an instance of the general problem with $g := \|\cdot - a\|$ and $h := \delta_C$ the indicator function of C . Indeed, we have $\{h > 0\} = X \setminus C \neq X$, whereas $\{h \geq 0\} = X$. If a lies in the interior of C , then α is strictly positive; otherwise $\alpha = 0$.

Remark 2.2. The distance α may not be achieved on $X \setminus C$. On the other hand, if X is finite dimensional, α is achieved on the closure of $X \setminus C$, that is, the complement of the interior of C .

The problem of the distance to a reverse convex set is studied in detail elsewhere [1, 2] in the framework of a problem arising in quantitative economics. In this context,

the duality result stated below is used for deriving a global solution by reducing the problem to finitely many linear programs.

Example 2.2 (D.C. optimization). Let X, g, h be as in the beginning of this section. We are interested in finding

$$\alpha := \inf_{x \in X} \{g(x) - h(x)\}$$

or, in short,

$$\alpha := \inf_X \{g - h\}.$$

In order to set this problem in the general framework, let us introduce, on the product space $\tilde{X} := X \times \mathbb{R}$, the proper convex functions $\tilde{g}$ and $\tilde{h}$ as follows:

$$\tilde{g}(x, \xi) := g(x) - \xi, \quad \tilde{h}(x, \xi) := h(x) - \xi.$$

PROPOSITION 2.1. $\alpha = \inf_{\tilde{h} > 0} \tilde{g} = \inf_{\tilde{h} > 0} \tilde{g}$.

Proof. For all x in X and ξ in $\mathbb{R}$ such that $h(x) \geq \xi$, we have

$$\alpha \leq g(x) - h(x) \leq g(x) - \xi.$$

Therefore,

$$\alpha \leq \inf_{\tilde{h} \geq 0} \tilde{g} \leq \inf_{\tilde{h} > 0} \tilde{g} \leq g(x) - \xi, \quad \text{for all } x \in X, \xi \in \mathbb{R}, h(x) > \xi.$$

Keeping x fixed and making ξ tend to $h(x)$, we get

$$\alpha \leq \inf_{\tilde{h} \geq 0} \tilde{g} \leq \inf_{\tilde{h} > 0} \tilde{g} \leq g(x) - h(x), \quad \text{for all } x \in X. \quad \square$$

3. Duality. Examples.

Example 3.1. Let $(X, \|\cdot\|)$ be a real normed linear space. We denote $\|\cdot\|_*$ the dual norm on the dual space Y . Let C be a nonempty closed convex subset in X and $a \in X$. Let us recall that the (primal) problem is to find $\alpha := d(a, X \setminus C)$.

DEFINITION 3.1. *The dual problem is to find β :*

$$\beta := \inf_{\|y\|_* = 1} \{\delta_C^*(y) - \langle y, a \rangle\},$$

where δ_C^* denotes the Legendre–Fenchel conjugate of the indicator function of C , that is, the support function of C .

PROPOSITION 3.1. *If $a \in C$, then the infimum in the dual problem can also be taken on the complement of the open or closed dual unit ball, without changing β .*

Proof. By translation we can assume $a = 0$ since

$$d(a, X \setminus C) = d(0, X \setminus C - a) = d(0, X \setminus (C - a))$$

and

$$\delta_{C-a}^* = \delta_C^* - \langle a, \cdot \rangle.$$

If $C = X$, the three quantities (as well as α) are equal to $+\infty$. So we can assume $C \neq X$.

Let y in Y be such that $\|y\|_* \geq 1$. Then $\frac{1}{\|y\|_*} \leq 1$ and, as $0 \in C$, $\delta_C^* \geq 0$. Therefore,

$$\inf_{\|z\|_*=1} \delta_C^*(z) \leq \delta_C^*\left(\frac{y}{\|y\|_*}\right) = \frac{\delta_C^*(y)}{\|y\|_*} \leq \delta_C^*(y)$$

and

$$\inf_{\|y\|_*=1} \delta_C^*(y) = \inf_{\|y\|_* \geq 1} \delta_C^*(y).$$

Moreover, $\text{dom } \delta_C^*$, the effective domain of the support function of C , is a convex cone different from 0 because C is different from X . Therefore it contains a half-line with origin 0 generated by some $\bar{y}$ on the dual unit sphere. Let $\bar{\lambda} > 1$. By convexity, we have, for all λ greater than one and lesser than $\bar{\lambda}$,

$$\inf_{\|y\|_* > 1} \delta_C^*(y) \leq \delta_C^*(\lambda \bar{y}) \leq \delta_C^*(\bar{y}) + \frac{(\lambda - 1)}{(\bar{\lambda} - 1)} (\delta_C^*(\bar{\lambda} \bar{y}) - \delta_C^*(\bar{y})).$$

Let λ tend to one in the above inequality. We get, in a concise writing,

$$\inf_{\|y\|_* > 1} \delta_C^*(y) = \inf_{\|y\|_* \geq 1} \delta_C^*(y) \leq \inf_{\|y\|_* > 1} \delta_C^*(y) \leq \inf_{\|y\|_* \geq 1} \delta_C^*(y) = \inf_{\|y\|_* \geq 1} \delta_C^*(y). \quad \square$$

Remark 3.1. If $a \in C$, then the dual problem appears also as an instance of the general problem, with the reverse convex constraint being strict or nonstrict without changing the infimum value.

The following proposition due to Franchetti–Singer ([5, Theorem 2.1]) states the duality result.

PROPOSITION 3.2. *If $a \in C$, then $\alpha = \beta$.*

Remark 3.2. (i) Of course, if $a \notin C$, then $\alpha = 0 > \beta$.

(ii) The assumption made in [5] that C is bounded is in fact useless in their proof.

(iii) As pointed out in [5], this duality result has to be compared with the duality formula for the distance to a closed convex set:

$$\text{for all } a \in X, \quad d(a, C) = \max_{\|y\|_* \leq 1} \{\langle a, y \rangle - \delta_C^*(y)\}.$$

A direct proof of this result can be found in [13]. It is also an easy consequence of the Fenchel–Moreau–Rockafellar formula giving the infimum of the sum of two convex functions [4, 12, 15, 16].

Furthermore, when $X \setminus C$ is itself convex, that is, C is the half-space $\{\langle y, \cdot \rangle \geq b\}$ with $y \in Y$, $y \neq 0$, $b \in \mathbb{R}$, and if $a \in C$, then both results reduce to the Ascoli formula:

$$d(a, X \setminus C) (= d(a, \overline{X \setminus C})) = \frac{\langle y, a \rangle - b}{\|y\|_*}.$$

(iii) Contrary to the case of the distance to a convex set, the infimum value in the dual problem may not be attained in the infinite-dimensional setting. More precisely, such achievement cannot be obtained directly by compactness arguments since, though the function to be minimized ($\delta_C^* - \langle \cdot, a \rangle$) is lower semicontinuous and convex so it is weak star lower semicontinuous, the dual unit sphere is not weak star

compact in general. Nevertheless, it is shown in [5, Theorem 4.3a], that if the set of nearest points is not empty, then the infimum in the dual problem is attained.

Example 3.2. The duality result for D.C. optimization we deal with is well known and is due to Toland [25] and Singer [17].

PROPOSITION 3.3. *Let X, g, h be as in the beginning of section 2. If h is closed, i.e., proper and lower semicontinuous or the constant $+\infty$ or $-\infty$, then*

$$\alpha := \inf_{x \in X} \{g(x) - h(x)\} = \inf_{y \in Y} \{h^*(y) - g^*(y)\} =: \beta,$$

where $\star$ denotes the Legendre–Fenchel conjugacy.

Let us recall that the (very simple) proof uses only the property $h = h^{**}$ (in particular g need not be convex) and the “inf inf” exchange principle.

4. Duality. General case. Let X, g, h be as in the beginning of section 2. Let us recall that the (primal) problem is to find $\alpha := \inf_{h>0} g$.

DEFINITION 4.1. *The dual problem is to find β :*

$$\beta := \inf_{y \in Y(g, h)} \sup_{\lambda \geq 0} \{\lambda h^*(y) - g^*(\lambda y)\},$$

where $Y(g, h) := \{y \in Y; \text{dom } g \cap \{\langle y, \cdot \rangle > h^*(y)\} \neq \emptyset\}$, the supremum on $\lambda \geq 0$ being attained for all $y \in Y(g, h)$ as shown in the proof of Theorem 4.1 below.

Remark 4.1. Clearly, $Y(g, h)$ is contained in $\text{dom } h^*$. Moreover, as pointed out to us by M. Volle, professor at the Université d’Avignon, France, $Y(g, h)$ can be described by means of a D.C. constraint. More precisely, we have

$$Y(g, h) = \{y \in Y; h^*(y) - \delta_{\text{dom } g}^*(y) < 0\}.$$

Indeed, $y \notin Y(g, h)$ if and only if $\text{dom } g \cap \{\langle y, \cdot \rangle > h^*(y)\} = \emptyset$, i.e., $\text{dom } g \subset \{\langle y, \cdot \rangle \leq h^*(y)\}$, i.e., $\delta_{\text{dom } g}^*(y) \leq h^*(y)$.

Now we can state the main duality result.

THEOREM 4.1. *Let X, g, h be as in the beginning of section 2. If h is closed, then $\alpha = \beta$.*

Proof. First of all, as $h = h^{**}$, we note that

$$(4.1) \quad \{h > 0\} = \cup_{y \in Y} \{\langle y, \cdot \rangle > h^*(y)\}.$$

Let $\alpha(y) := \inf\{g(x); \langle y, x \rangle > h^*(y)\}$. If $y \notin Y(g, h)$, then $\alpha(y) = +\infty$; otherwise $-\infty \leq \alpha(y) < +\infty$. Therefore,

$$(4.2) \quad \alpha = \inf_{y \in Y} \alpha(y) = \inf_{y \in Y(g, h)} \alpha(y)$$

for all $y \in \text{dom } h^*$, $\alpha(y) \geq \overline{\alpha(y)} := \inf_{\langle y, \cdot \rangle \geq h^*(y)} g = \inf_{x \in X} \sup_{\lambda \geq 0} \{g(x) + \lambda(h^*(y) - \langle y, x \rangle)\}$

$$\geq \sup_{\lambda \geq 0} \left\{ \lambda h^*(y) + \inf_{x \in X} [g(x) - \langle \lambda y, x \rangle] \right\} =: \beta(y).$$

Here we used the convention stated in the introduction and the inequality “inf sup” $\geq$ “sup inf.”

It is worth noting that $\inf_{x \in X} \{g(x) - \langle \lambda y, x \rangle\} = -g^*(\lambda y)$. So, $\beta = \inf_{y \in Y(g, h)} \beta(y)$ and hence $\alpha \geq \beta$. It should be noticed that this last result does not require the convexity of g .

If $Y(g, h)$ is empty, then $\alpha = \beta = +\infty$. Otherwise (this being equivalent to feasibility: $\text{dom } g \cap \{h > 0\} \neq \emptyset$), let y in $Y(g, h)$. If $\alpha(y) = -\infty$, then $\alpha(y) = \beta(y) = -\infty$. If $\alpha(y) \in \mathbb{R}$, then, thanks to the convexity of g , $\alpha(y) = \overline{\alpha(y)}$. Then apply the “inf sup” theorem of Moreau ([11, Proposition 2 and its remark]) to the Lagrangian function $L : X \times \mathbb{R} \rightarrow \overline{\mathbb{R}}$ defined as follows:

$$L(x, \lambda) := \begin{cases} +\infty & \text{if } g(x) = +\infty \\ g(x) + \lambda(h^*(y) - \langle y, x \rangle) & \text{if } x \in \text{dom } g \\ -\infty & \text{if } x \in \text{dom } g \end{cases} \quad \begin{matrix} \text{and } \lambda \geq 0, \\ \text{and } \lambda < 0. \end{matrix}$$

So, as

$$\lim_{\lambda \rightarrow +\infty} g(x_0) + \lambda(h^*(y) - \langle y, x_0 \rangle) = -\infty$$

for all x_0 in $\text{dom } g \cap \{\langle y, \cdot \rangle > h^*(y)\}$, we get

$$\inf_{x \in X} \sup_{\lambda \in \mathbb{R}} L(x, \lambda) = \max_{\lambda \in \mathbb{R}} \inf_{x \in X} L(x, \lambda).$$

We conclude by noticing that

$$\overline{\alpha(y)} = \inf_{x \in X} \sup_{\lambda \in \mathbb{R}} L(x, \lambda)$$

and

$$\max_{\lambda \in \mathbb{R}} \inf_{x \in X} L(x, \lambda) = \max_{\lambda \geq 0} \{\lambda h^*(y) - g^*(\lambda y)\} = \beta(y).$$

Then, for all $y \in Y(g, h)$, $\alpha(y) = \beta(y)$ and, thanks to (4.2), $\alpha = \beta$. $\square$

Remark 4.2. (i) In a preliminary version, g was furthermore assumed to be proper closed, with the proof relying on the convex Lagrangian duality theorem ([3, Chap. III, Proposition 5.1]). The uselessness of this extra assumption was also pointed out to us by M. Volle. Actually, as a direct consequence of Theorem 4.1, if g admits a continuous affine minorant, then the primal infimum is the same with $\bar{g}$ closure of g , because $\bar{g}^* = g^*$ and hence the dual problem is the same.

(ii) Although it is not unreasonable to expect an analogous result with large inequality in the constraint ($h \geq 0$), the above technique of proof would fail in this case because relation (1), which is fundamental, is no more true in general with large inequality on both sides unless (for instance) h is finite valued and continuous. Moreover, even with this extra assumption, one could not continue as above.

(iii) Concerning the connection of our Theorem 4.1 with existing general results on reverse convex duality, we observe that Theorem 4.1 of [20] yields, with our notations, the following: let X be a real, locally convex space, $g : X \rightarrow \overline{\mathbb{R}}$, C a closed convex set in X . Then

$$\inf_{x \notin C} g(x) = \inf_{y \in \text{dom } \delta_C^* \setminus \{0\}} \alpha(y),$$

which is exactly the same as (4.2) above (recall that at this stage of the proof of our Theorem 4.1 the convexity of g was not required) in the particular case $h := \delta_C$

since, in this case, $Y(g, h) \subset \text{dom } \delta_C^* \setminus \{0\}$ and if $y \in \text{dom } \delta_C^* \setminus \{0\}$, but $y \notin Y(g, h)$, then $\alpha(y) = +\infty$. So, even in this particular case $h := \delta_C$, the condition on the dual variable y is more precise in our result, and this allows us to go farther in the duality when g is convex: express $\alpha(y)$ in terms of the Legendre–Fenchel conjugates of g and δ_C .

COROLLARY 4.1. *Let us assume $\text{dom } g = X$. If $\inf h \leq 0$, then*

$$\alpha = \inf_{y \in \text{dom } h^* \setminus \{0\}} \sup_{\lambda \geq 0} \{\lambda h^*(y) - g^*(\lambda y)\}.$$

If $\inf h > 0$, the duality result reduces to $\alpha = -g^*(0)$.

Proof. Actually, $Y(g, h) = \{y \in Y; \{\langle y, \cdot \rangle > h^*(y)\} \neq \emptyset\} = \text{dom } h^* \setminus \{0\}$ if $h^*(0) = -\inf h \geq 0$, $\text{dom } h^*$ if $h^*(0) < 0$.

If $-\inf h = h^*(0) < 0$, then $\{h > 0\} = X$. Therefore, $\alpha = \inf_X g = -g^*(0)$.

On the other hand, for all $y \in \text{dom } h^*$, $\beta(y) \geq -g^*(0) = \beta(0) = \sup_{\lambda \geq 0} \{\lambda h^*(0) - g^*(0)\}$. Therefore, $\beta = \inf_{y \in \text{dom } h^*} \beta(y) = -g^*(0)$. $\square$

5. Coming back to the examples.

Example 5.1. For the sake of simplicity let us assume $a = 0$. We are in the framework of Corollary 4.1, with $g = \|\cdot\|$, $h = \delta_C$ and, therefore, $g^* = \delta_{B_*}$, where B_* denotes the dual unit ball. Applying Corollary 4.1 (case $\inf h \leq 0$) we get

$$d(0, X \setminus C) = \inf_{y \in \text{dom } \delta_C^* \setminus \{0\}} \sup_{\lambda \geq 0} \{\lambda \delta_C^*(y) - \delta_{B_*}(\lambda y)\}.$$

Now, the supremum in the right-hand side is equal to

$$\sup\{\lambda \delta_C^*(y); \lambda \geq 0, \|\lambda y\|_* \leq 1\}$$

which, for $y \neq 0$, is equal to

$$\sup\left\{\lambda \delta_C^*(y); 0 \leq \lambda \leq \frac{1}{\|y\|_*}\right\} = \begin{cases} \frac{1}{\|y\|_*} \delta_C^*(y) & \text{if } \delta_C^*(y) \geq 0, \\ 0 & \text{if } \delta_C^*(y) < 0. \end{cases}$$

Finally,

$$d(0, X \setminus C) \geq \inf_{\|y\|_* = 1} \delta_C^*(y)$$

with equality if and only if $0 \in C$, that is, we recover Proposition 3.2.

Example 5.2. With the same notation as in section 2, we have

$$\alpha := \inf_X \{g - h\} = \inf_{\tilde{h}(x, \xi) > 0} \tilde{g}(x, \xi).$$

Let x be in $\text{dom } g$ and $\xi \in \mathbb{R}$, $\xi < h(x)$; then (x, ξ) is in $\text{dom } \tilde{g}$ ($= \text{dom } g \times \mathbb{R} \cap \{\tilde{h} > 0\}$). So, feasibility mentioned in the proof of Theorem 4.1 is fulfilled.

By a simple calculation, we get $\tilde{h}^*$:

$$\tilde{h}^*(y, \eta) = \begin{cases} h^*(y) & \text{if } \eta = -1, \\ +\infty & \text{otherwise.} \end{cases}$$

Analogue holds true for $\tilde{g}^*$.

$$\tilde{Y}(\tilde{g}, \tilde{h}) = \{(y, -1); \exists x \in \text{dom } g, \exists \xi \in \mathbb{R}, \langle y, x \rangle - \xi > h^*(y)\} = \text{dom } h^* \times \{-1\}.$$

For $y \in \text{dom } h^*$ and $\eta = -1$, we have

$$\lambda \tilde{h}^*(y, \eta) - \tilde{g}^*(\lambda y, \lambda \eta) = \begin{cases} h^*(y) - g^*(y) & \text{if } \lambda = 1, \\ -\infty & \text{otherwise.} \end{cases}$$

If h is closed, then $\tilde{h}$ is also closed. So, applying Theorem 4.1, we get (with the usual convention recalled in the introduction),

$$\alpha = \inf_{y \in \text{dom } h^*} \{h^*(y) - g^*(y)\} = \inf_Y \{h^* - g^*\}.$$

Remark 5.1. (i) Actually, in the theorem of Toland–Singer, g need not be convex. Now, in the above derivation, g is assumed to be proper convex. Of course, applying the theorem of Toland–Singer twice we get

$$\inf_X \{g - h\} = \inf_X \{g^{**} - h\}.$$

On the other hand, for any g (and h proper closed convex), it can be easily proved that

$$\inf_X \{g - h\} = \inf_X \{\overline{\text{co}}g - h\},$$

where $\overline{\text{co}}g$ denotes the closed convex hull of g , that is, the function whose epigraph is the closed convex hull of the epigraph of g . Therefore, by our derivation we get, for any g and proper closed convex h ,

$$\inf_X \{g - h\} = \inf_Y \{h^* - (\overline{\text{co}}g)^*\}.$$

In fact, our derivation assumes $\overline{\text{co}}g$ proper. In the other cases ($\overline{\text{co}}g$ taking the value $-\infty$ or $\overline{\text{co}}g$ identically $+\infty$) the equality is obtained directly, with the usual convention. Finally, as $(\overline{\text{co}}g)^* = g^*$ (possibly identically $+\infty$), from our general duality result we can recover the theorem of Toland–Singer for any g .

(ii) Conversely, it does not seem possible to deduce the main result of section 4 from the Toland–Singer’s formula or, equivalently, from the general formula giving the Legendre–Fenchel conjugate of a difference of functions, the second one being closed convex [7], because reverse convex optimization, in a general setting, is not known to be reducible to the minimization of a difference of functions, the second one being closed convex. As shown in [22], a reduction to the minimization of a difference of functions is possible for a subclass of reverse convex optimization problems (for which $X = Y = \mathbb{R}^n$) via a duality scheme relying on an operation on functions called quasi-conjugacy. Unfortunately, in the difference to be minimized, the second function may be nonclosed convex and so, the Toland–Singer’s formula cannot be applied.

Acknowledgments. The author would like to thank the anonymous referees for their valuable remarks and suggestions.

REFERENCES

- [1] W. BRIEC, *Une nouvelle approche de la mesure de l’efficacité productive*, Thèse de Doctorat en Sciences Economiques, Ecole des Hautes Etudes en Sciences Sociales, Aix-Marseille, 1994.
- [2] W. BRIEC AND B. LEMAIRE, *Technical efficiency and distance to a reverse convex set*, European J. Oper. Res., to appear.

- [3] I. Ekeland and R. Temam, *Convex Analysis and Variational Problems*, North-Holland American Elsevier, New York, 1976.
- [4] W. Fenchel, *Convex Cones, Sets and Functions*, Multilith Lecture Notes, Princeton University, Princeton, NJ, 1953.
- [5] C. Franchetti and I. Singer, *Best approximation by elements of caverns in normed linear spaces*, Boll. Un. Mat. Ital. B (5), 17 (1980), pp. 33–43.
- [6] J.-B. Hiriart-Urruty, *Generalized differentiability, duality and optimization for problems dealing with differences of convex functions*, Lecture Notes in Econom. and Math. Systems 256, J. Ponstein, ed., Springer-Verlag, Berlin, 1985, pp. 37–70.
- [7] J.-B. Hiriart-Urruty, *A general formula on the conjugate of the difference of functions*, Canad. Math. Bull., 29 (1986), pp. 482–485.
- [8] J. B. Hiriart-Urruty, *From convex optimization to nonconvex optimization. Necessary and sufficient conditions for global optimality*, in Nonsmooth Optimization and Related Topics (Erice 1988), F. H. Clarke, V. F. Demyanov, and F. Gianessi, eds., Plenum Press, New York, 1989, pp. 219–239.
- [9] J. B. Hiriart-Urruty, *How to regularize a difference of convex functions*, J. Math. Anal. Appl., 162 (1991), pp. 196–209.
- [10] R. Horst and H. Tuy, *Global Optimization (Deterministic Approach)*, 3rd ed., Springer-Verlag, New York, 1996.
- [11] J. J. Moreau, *Théorèmes inf-sup*, C. R. Acad. Sci. Paris, 258 (1964), pp. 2720–2722.
- [12] J. J. Moreau, *Fonctionnelles convexes*, Séminaire sur les e.d.p., Collège de France, 1966.
- [13] L. Nirenberg, *Functional Analysis*, New York University, New York, 1961.
- [14] P. D. Tao and E. B. Souad, *Duality in D.C. optimization. Subgradient methods*, in Trends in Mathematical Optimization, Internat. Ser. Numer. Math. 84(c), Birkhäuser Verlag, Basel, 1988, pp. 277–293.
- [15] R. T. Rockafellar, *Duality theorems for convex functions*, Bull. Amer. Math. Soc., 70 (1964), pp. 189–192.
- [16] R. T. Rockafellar, *Extension of Fenchel's duality theorem for convex functions*, Duke Math. J., 33 (1966), pp. 81–90.
- [17] I. Singer, *A Fenchel-Rockafellar type duality theorem for maximization*, Bull. Austral. Math. Soc., 20 (1979), pp. 193–198.
- [18] I. Singer, *Maximization of lower semicontinuous convex functionals on bounded subsets of locally convex spaces. II: Quasi-Lagrangian duality theorems*, Results. Math., 3 (1980), pp. 235–248.
- [19] I. Singer, *Minimization of continuous convex functionals on complements of convex subsets of locally convex spaces*, Math. Oper. Statist. Ser. Optim., 11 (1980), pp. 221–234.
- [20] I. Singer, *Optimization by level set methods, VI: Generalization of surrogate type reverse convex duality*, Optimization, 18 (1987), pp. 485–499.
- [21] I. Singer, *Some further duality theorems for optimization problems with reverse convex constraint sets*, J. Math. Anal. Appl., 171 (1992), pp. 205–219.
- [22] P. T. Thach, *Quasiconjugates of functions. Duality relationship between quasiconvex minimization under a reverse convex constraint and quasiconvex maximization under a convex constraint, and applications*, J. Math. Anal. Appl., 159 (1991), pp. 299–322.
- [23] P. T. Thach, R. E. Burkard, and W. Oettli, *Mathematical programs with a two-dimensional reverse convex constraint*, J. Global Optim., 1 (1991), pp. 145–154.
- [24] P. T. Thach, *A nonconvex duality with zero gap and applications*, SIAM J. Optim., 4 (1994), pp. 44–64.
- [25] J. F. Toland, *A duality principle for nonconvex optimization and the calculus of variation*, Arch. Rational Mech. Anal., 71 (1979), pp. 41–61.
- [26] H. Tuy, *D.C. optimization: Theory, methods and algorithms*, in Handbook of Global Optimization, R. Horst and P. M. Pardalos, eds., Kluwer Academic Publishers, Norwell, MA, 1995, pp. 149–216.

A NEW DEGENERACY METHOD AND STEEPEST-EDGE-BASED CONDITIONING FOR LP*

ROGER FLETCHER†

Abstract. A new recursive method for resolving degeneracy in simplex-like methods for linear programming (LP) is described. The method provides a guarantee of termination, even in the presence of round-off errors, and is readily implemented. In contrast to a previous method of the author, this method works throughout in the primal space. One consequence is that the steepest-edge criterion can be used on all iterations and at all levels of recursion.

It is also shown that the associated steepest-edge coefficients provide information from which the expected condition of the current LP basis can be calculated cheaply. This provides a more accurate indication of the actual condition of a system than is obtained from norm-based condition numbers. This idea also enables the condition of null space matrices to be estimated.

Key words. steepest-edge method, degeneracy, linear programming, conditioning, simplex method, active set method, null space

AMS subject classifications. 65K, 90C

PII. S1052623494277470

1. Introduction. Steepest-edge methods were introduced by Goldfarb and Reid [9] and Greenberg and Kalan [10] to improve the performance of the primal simplex method for solving LP problems of the form

$$(1.1) \quad \begin{array}{ll} \text{minimize} & f = c^T x, \quad x \in \mathbb{R}^n, \\ \text{subject to} & Ax = b, \quad x \geq 0. \end{array}$$

Similar ideas are applicable to the solution of an LP problem by an active set method (ASM); for example, steepest-edge variants of the dual simplex method have been fully explored by Forrest and Goldfarb [8]. Certain coefficients must be updated at each iteration to normalize edges of the feasible region polytope. Experimental evidence indicates that the updating cost is generally outweighed by a reduction in the number of iterations required.

Degeneracy causes difficulty for simplex-like methods when it blocks progress in the current iteration. Often the problem is transformed so that application of a simplex-like method can be used to remove the blockage. In some cases a considerable number of iterations are required to do this. It is therefore beneficial in such cases if advantage can be taken of the steepest-edge approach during this procedure.

The author has been interested in a particular primal-dual method [5] for resolving degeneracy. A feature of the method is that it permits the use of heuristics that enable termination to be guaranteed in the presence of arbitrary round-off errors. The property is achieved as a consequence of the recursive solution of *feasibility problems* and the reasons are reviewed in section 2. A particular method for phase I of the simplex method plays an important part and is also described. The method is implemented in a quadratic programming (QP) code `bqpd`.

*Received by the editors November 21, 1994; accepted for publication (in revised form) September 15, 1997; published electronically September 23, 1998. A previous version of this paper was presented at the 15th International Symposium on Mathematical Programming, Ann Arbor, MI, August 15–19, 1994.

<http://www.siam.org/journals/siopt/8-4/27747.html>

†Department of Mathematics, University of Dundee, Dundee DD1 4HN, Scotland, UK (fletcher@mcs.dundee.ac.uk).

The work in this paper came about by considering how best to incorporate the steepest-edge approach into **bqpd**. A feature of the method [5] used in **bqpd** is that the recursively generated problems alternate between the primal and dual spaces of the LP. If the steepest-edge criterion were to be used on every iteration, steepest-edge coefficients would be needed in both spaces, which would considerably increase the effort required to update these coefficients. The alternative of using the steepest-edge criterion only in the primal space is likely to be more efficient but does not take full advantage of the steepest-edge approach.

The main aim of this paper is to describe a new method for resolving degeneracy that is also based on the recursive solution of feasibility problems and thus provides a similar guarantee of termination. The method works in the primal space throughout, so that the steepest-edge criterion can be used on every iteration, and only primal steepest-edge coefficients need to be maintained. The method can be extended to solve quadratic programming problems by following the idea described in [6].

The new method is described in section 2 in the context of the traditional form of the simplex method. In section 3, the new method is compared with that in [5] and with Wolfe's method [13]. It is argued that Wolfe's method, which also works only in the primal space, does not have the same guarantee of termination for inexact arithmetic.

In section 4 the ASM is described and is shown to follow a different path to the solution of (1.1) than the primal simplex method, when the steepest-edge criterion is used. It is shown how the new method for resolving degeneracy can be applied in an ASM context. Details of implementation are also discussed, and it is shown that the new method is just as convenient in this respect as the method in [5].

Some numerical experiments with a steepest-edge ASM on the SOL test problem set are also described in section 4. These confirm that the steepest-edge approach gives a significant improvement in performance. They also show that the new method for degeneracy provides a viable alternative to that used in **bqpd**. However, the proportion of dual iterations taken by **bqpd** is relatively small, so the opportunity for improving overall performance by using the steepest-edge criterion at all levels is limited.

Given that steepest-edge coefficients are maintained by the algorithm, it is worthwhile to consider if they can be used in any other heuristics in an LP code. Difficulties often arise when the basis matrix in a simplex-type method becomes nearly singular. It is useful to be able to assess the condition of this matrix, so that a *basis repair* technique can be initiated when necessary. It is important that condition estimates should not be too pessimistic, for otherwise basis repair would be invoked too frequently. This paper shows that the availability of steepest-edge coefficients enables a realistic estimate to be made of the condition of systems of equations involving the basis matrix.

The amount of additional computation is small, particularly for a sparse LP problem. The idea also permits the condition of a null space matrix to be estimated, with application to reduced Hessian methods for quadratic programming and nonlinear programming.

2. A new method for resolving degeneracy. In this section a new method is described for resolving degeneracy in the traditional simplex method (Dantzig [2]), which considers a primal LP problem in the form

$$(2.1) \quad \begin{array}{ll} \text{minimize} & f = c^T x, \quad x \in \mathbb{R}^n, \\ \text{subject to} & Ax = b, \quad x \geq 0, \end{array}$$

where $A \in \mathbb{R}^{m \times n}$. The new method is based on the recursive solution of *feasibility problems*, analogous to the problem of finding a feasible point for (2.1) (phase I of the simplex method). There are various ways of doing this and my practical experience does not indicate any strong preferences. However, in the context of this paper, one method is particularly useful and might be described as the *one-at-a-time method*. This version of the simplex method is outlined in the first part of this section. The new method for resolving degeneracy is then described, and some of its properties are discussed. The method is compared with related methods in section 3.

In the simplex method applied to (2.1), the variables x are partitioned into basic variables x_B and nonbasic variables x_N , indexed by the sets B and N , respectively. Correspondingly, A is partitioned as $A = [A_B \mid A_N]$, where A_B is an $m \times m$ nonsingular matrix, and c is partitioned as $c^T = [c_B^T, c_N^T]$. Then the constraints in (2.1) are equivalent to the system

$$(2.2) \quad x_B + \hat{A}x_N = \hat{x}_B, \quad x_B \geq 0, \quad x_N \geq 0,$$

where $\hat{A} = A_B^{-1}A_N$ and $\hat{x}_B = A_B^{-1}b$. Such a problem is referred to as a *feasibility problem*. The current value of x in the simplex method is denoted by $\hat{x}$, with $\hat{x}_N = 0$ and $\hat{x}_B = A_B^{-1}b$ as above. It is possible to write the equations in (2.2) as

$$(2.3) \quad x_B = \hat{x}_B - \hat{A}x_N,$$

in which the basic variables are regarded as being dependent on the nonbasic variables. It is convenient to refer to rows and columns of $\hat{A}$ by the variables to which they correspond.

A feature of the simplex method is that, after each iteration, an index $q \in B$ is interchanged with an index $p \in N$. It is readily possible to rearrange (2.2) to correspond with the new definitions of B and N , by making row operations in (2.2) that transform column p of $\hat{A}$ to a unit vector with a 1 in row q , followed by an exchange of columns so that this unit vector becomes part of x_B . This is referred to as *pivoting* on $\hat{a}_{qp}$. In practice $\hat{A}$ is not explicitly stored and updated, but relevant rows and columns are generated as required by solving systems involving A_B . It is therefore an invertible representation of A_B that is updated. However, the format of (2.2) and (2.3) is often valuable for explaining how a method works.

A brief description of this version of the simplex method is now given. If $\hat{x}_B \geq 0$, then $\hat{x}$ is feasible in (2.2) and hence (2.1). Otherwise (phase I) feasibility can be restored by the one-at-a-time method as follows. The idea is to pick a *target variable* x_q , $q \in B$, for which $\hat{x}_q < 0$, and to drive x_q up to zero without creating any new infeasible basic variables. If the entries in row q of $\hat{A}$ are nonnegative, then it follows from (2.3) and $x_N \geq 0$ that no increase in x_q is possible and hence (2.1) has no solution. Otherwise some $p \in N$ is chosen such that $\hat{a}_{qp} < 0$ (traditionally the most negative value of $\hat{a}_{qp}$). This is referred to as the *optimality test*. Then (in absence of degeneracy) x_q can be increased by increasing x_p . The effect on the basic variables is now considered. As x_p is increased, it follows from (2.3) that any x_i , $i \in B$, becomes zero when $x_p = \alpha_i$, where α_i is defined by $\alpha_i = \hat{x}_i / \hat{a}_{ip}$. Let $q' \in B$ be the index of the first nonnegative basic variable to become zero, that is, the index that attains

$$(2.4) \quad \min \alpha_i, \quad i \in B, \quad \hat{x}_i \geq 0, \quad \hat{a}_{ip} > 0.$$

This is referred to as the *ratio test*. If q' does not exist, or if $\alpha_{q'} \geq \alpha_q$, then increasing x_p to α_q causes x_q to be increased to zero without introducing any new infeasibility

in $\hat{x}_B$. Thus the infeasibility in x_q is removed and a pivot can be taken on $\hat{a}_{qp}$.¹ Otherwise $x_{q'}$ is reduced to zero, a pivot on $\hat{a}_{q'p}$ is made, and the attempt to increase x_q is repeated. Eventually feasibility is restored to x_q . The procedure is repeated until no infeasibility remains (unless it is found that (2.1) has no solution).

Once a feasible point is located, a similar method is used to minimize the objective function $f = c^T x$ (phase II). Setting $x_0 = -f$, the equation

$$(2.5) \quad x_0 = -\hat{f} - \hat{c}^T x_N$$

is obtained by using (2.3) to eliminate x_B from f , where $\hat{f} = c_B^T \hat{x}_B$ is the current value of f , and where $\hat{c} = c_N - \hat{A}^T c_B$. It is convenient to regard (2.5) as an additional row of the system (2.3), with $\hat{c}^T$ as row 0 of $\hat{A}$ and x_0 as the target variable to be increased. The method of the previous paragraph is then applied (with $q = 0$ and allowing x_0 to increase without restriction) either until optimality is detected or until an unbounded increase in x_0 occurs. In fact it is more satisfactory for guaranteeing termination for inexact arithmetic if the user supplies a lower limit $f_{\min}$ on f , and the code terminates when x_0 would increase to $-f_{\min}$.

However, it may be that the increase in x_p is *blocked* because of degeneracy; that is, the minimum ratio in (2.4) is $\alpha_{q'p} = 0$. Thus no increase in x_q is obtained. Although a pivot on $\hat{a}_{q'p}$ may be taken, it is possible that the algorithm could cycle infinitely and fail to find a solution.

The new method for resolving degeneracy in the simplex method is now introduced. It is perhaps most readily understood by following through an example. Consider the LP

$$(2.6) \quad \begin{array}{ll} \text{minimize} & -\frac{3}{4}x_1 + 20x_2 - \frac{1}{2}x_3 + 6x_4 \\ \text{subject to} & \frac{1}{4}x_1 - 8x_2 - x_3 + 9x_4 \leq 0, \\ & \frac{1}{2}x_1 - 12x_2 - \frac{1}{2}x_3 + 3x_4 \leq 0, \\ & x_3 \leq 1, \quad x_i \geq 0, \quad i = 1, 2, 3, 4, \end{array}$$

which was devised by Beale to illustrate cycling in the simplex method. Slack variables x_5, x_6, x_7 are added to the first three constraints. The initial realization of (2.3), including the target variable x_0 for the objective function as defined in (2.5), is

$$(2.7) \quad \begin{pmatrix} x_0 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} - \begin{bmatrix} -\frac{3}{4} & 20 & -\frac{1}{2} & 6 \\ \frac{1}{4} & -8 & -1 & 9 \\ \frac{1}{2} & -12 & -\frac{1}{2} & 3 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}$$

and is feasible so that no phase I is required. In phase II, x_0 is the target variable to increase. In the optimality test, the most negative entry $-\frac{3}{4}$ in the x_0 row is chosen, indicating that the nonbasic variable x_1 is to be increased. However, the resulting ratio test is blocked by both x_5 and x_6 .

The first stage in resolving the degeneracy block is to ignore the nondegenerate x_7 row and replace $x_0 - \hat{x}_0$ by δ_0 (in this example $\hat{x}_0$, the current value of x_0 , is zero

¹It is also possible here to look for a larger step that removes more infeasibilities.

and $\delta_0 = x_0$). This gives rise to the system

$$(2.8) \quad \begin{pmatrix} \delta_0 \\ x_5 \\ x_6 \end{pmatrix} = - \begin{bmatrix} -\frac{3}{4} & 20 & -\frac{1}{2} & 6 \\ \frac{1}{4} & -8 & -1 & 9 \\ \frac{1}{2} & -12 & -\frac{1}{2} & 3 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}, \quad x_i \geq 0, \quad i = 1, \dots, 6.$$

Any nontrivial solution of this system provides a feasible direction for (2.7) from the current point $\hat{x}$. However, we need to find a feasible direction that also increases δ_0 . We do this by finding a normalized solution of (2.8) for which $\delta_0 = 1$. This may be done in a convenient way by first pivoting on the element $-\frac{3}{4}$ selected in the optimality test, giving

$$(2.9) \quad \begin{pmatrix} x_1 \\ x_5 \\ x_6 \end{pmatrix} = - \begin{bmatrix} -\frac{4}{3} & -\frac{80}{3} & \frac{2}{3} & -8 \\ \frac{1}{3} & -\frac{4}{3} & -\frac{7}{6} & 11 \\ \frac{2}{3} & \frac{4}{3} & -\frac{5}{6} & 7 \end{bmatrix} \begin{pmatrix} \delta_0 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}, \quad x_i \geq 0, \quad i = 1, \dots, 6,$$

and then setting $\delta_0 = 1$, giving rise to the feasibility problem

$$(2.10) \quad \begin{pmatrix} x_1 \\ x_5 \\ x_6 \end{pmatrix} = \begin{pmatrix} \frac{4}{3} \\ -\frac{1}{3} \\ -\frac{2}{3} \end{pmatrix} - \begin{bmatrix} -\frac{80}{3} & \frac{2}{3} & -8 \\ -\frac{4}{3} & -\frac{7}{6} & 11 \\ \frac{4}{3} & -\frac{5}{6} & 7 \end{bmatrix} \begin{pmatrix} x_2 \\ x_3 \\ x_4 \end{pmatrix}, \quad x_i \geq 0, \quad i = 1, \dots, 6.$$

An attempt is now made to solve (2.10) by the one-at-a-time method. Note that the initial values of x_5 and x_6 are infeasible, which is a consequence of the degeneracy block in (2.7).

The largest infeasibility in (2.10) is $\hat{x}_6 = -\frac{2}{3}$, which determines x_6 as the target variable. The negative element $\hat{a}_{63}$ indicates that x_6 can be increased by increasing x_3 . A ratio test using the x_3 column of $\hat{A}$ indicates that x_6 reaches zero before x_1 , so that a pivot can be taken on $\hat{a}_{63}$. This gives rise to the system

$$(2.11) \quad \begin{pmatrix} x_1 \\ x_5 \\ x_3 \end{pmatrix} = \begin{pmatrix} \frac{4}{5} \\ \frac{3}{5} \\ \frac{4}{5} \end{pmatrix} - \begin{bmatrix} -\frac{128}{5} & \frac{4}{5} & -\frac{12}{5} \\ -\frac{16}{5} & -\frac{7}{5} & \frac{6}{5} \\ -\frac{8}{5} & -\frac{6}{5} & -\frac{42}{5} \end{bmatrix} \begin{pmatrix} x_2 \\ x_6 \\ x_4 \end{pmatrix}.$$

This step has also removed the infeasibility in x_5 , so a feasible solution to (2.8) has been obtained with $\delta_0 = 1$, $x_1 = x_3 = \frac{4}{5}$, $x_5 = \frac{3}{5}$, and $x_2 = x_4 = x_6 = 0$. If the same pivots are made in (2.7) (after defining $\delta_0 = x_0 - \hat{x}_0$), the outcome is

$$(2.12) \quad \begin{pmatrix} x_1 \\ x_5 \\ x_3 \\ x_7 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} - \begin{bmatrix} -\frac{4}{5} & -\frac{128}{5} & \frac{4}{5} & -\frac{12}{5} \\ -\frac{3}{5} & -\frac{16}{5} & -\frac{7}{5} & \frac{6}{5} \\ -\frac{4}{5} & -\frac{8}{5} & -\frac{6}{5} & -\frac{42}{5} \\ \frac{4}{5} & \frac{8}{5} & \frac{6}{5} & \frac{42}{5} \end{bmatrix} \begin{pmatrix} \delta_0 \\ x_2 \\ x_6 \\ x_4 \end{pmatrix},$$

and it can be seen that the right-hand side vector in (2.11) is the negative of the degenerate part of the δ_0 column of $\hat{A}$ in (2.12). Any of these elements that are nonzero can be used to pivot δ_0 back into B and thus return the system to standard form. Here we choose the (x_1, δ_0) element as pivot, giving rise to the primal system

$$(2.13) \quad \begin{pmatrix} x_0 \\ x_5 \\ x_3 \\ x_7 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} - \begin{bmatrix} -\frac{5}{4} & 32 & -1 & 3 \\ -\frac{3}{4} & 16 & -2 & 3 \\ -1 & 24 & -2 & -6 \\ 1 & -24 & 2 & 6 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_6 \\ x_4 \end{pmatrix}.$$

It can be seen that there is no longer a degeneracy block and x_1 can be increased. The resulting pivot on $\hat{a}_{71}$ gives rise to the system

$$(2.14) \quad \begin{pmatrix} x_0 \\ x_5 \\ x_1 \\ x_3 \end{pmatrix} = \begin{pmatrix} \frac{5}{4} \\ \frac{3}{4} \\ 1 \\ 1 \end{pmatrix} - \begin{bmatrix} \frac{3}{2} & 2 & \frac{5}{4} & \frac{21}{2} \\ -\frac{1}{2} & -2 & \frac{3}{4} & \frac{15}{2} \\ 2 & -24 & 1 & 6 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} x_6 \\ x_2 \\ x_7 \\ x_4 \end{pmatrix}.$$

Because the x_0 row is nonnegative, this is seen to provide the optimal solution of Beale's problem.

To illustrate a different outcome of the technique, let the (x_6, x_3) entry in (2.7) be changed from $-\frac{1}{2}$ to $\frac{1}{2}$ so that the corresponding entries in (2.9) and (2.10) change to $\frac{1}{6}$. Then there are no negative elements in the x_6 row of $\hat{A}$ in (2.10), indicating that x_6 cannot be increased and hence that the direction finding problem is infeasible. Hence $\hat{x} = 0$ would be an optimal solution of the modified version of (2.7). This is readily recognized by pivoting on the (x_6, δ_0) entry in (2.9) and replacing δ_0 by x_0 .

A more detailed description of the new method is now given. When there is no degeneracy (level 1), the method is just the traditional simplex method as outlined earlier in this section. Let q index the current target variable (in either phase I or phase II of the method), and let p be the index that is selected in the optimality test. If the ratio test (2.4) is blocked, x_q is replaced by the variable $\delta_q = x_q - \hat{x}_q$, thus making the right-hand side of the target row equal to zero. Then all the nondegenerate rows ($\hat{x}_i \neq 0$) in (2.3) are ignored, and the remaining *localized system*

$$(2.15) \quad \begin{aligned} \delta_q &= -\hat{a}_{q\bullet} x_N, \\ x_D &= -\hat{A}_D x_N, \quad x_D \geq 0, \quad x_N \geq 0, \end{aligned}$$

is considered, where $\hat{a}_{q\bullet}$ denotes row q of $\hat{A}$, x_D denotes the partition of x_B containing the degenerate basic variables ($\hat{x}_i = 0$), and $\hat{A}_D$ denotes the corresponding rows of $\hat{A}$. In order to find a feasible ascent direction for (2.2), we seek a solution of the localized system for which $\delta_q = 1$. To create a suitable feasibility problem, we proceed as in the example and pivot on the element $\hat{a}_{qp}$ in (2.15). Thus p is replaced by q in N , and p is added to the set D . This preserves the degeneracy present in (2.15) and gives a system that can be expressed as

$$(2.16) \quad x_D = -\hat{a}_q \delta_q - \sum_{i \in N \setminus q} \hat{a}_i x_i, \quad x_D \geq 0, \quad x_i \geq 0, \quad i \in N \setminus q,$$

where D , N , and $\hat{A}$ refer to the newly defined quantities that result from making the pivot, and $\hat{a}_i$ denotes column i of $\hat{A}$. Setting $\delta_q = 1$ then gives rise to the system

$$(2.17) \quad x_D = -\hat{a}_q - \sum_{i \in N \setminus q} \hat{a}_i x_i, \quad x_D \geq 0, \quad x_i \geq 0, \quad i \in N \setminus q.$$

This has essentially the same structure as (2.2) and is suitable for solution by the one-at-a-time method.

This system is regarded as a feasibility problem at level 2 of the recursive scheme. The initial values for solving (2.17) are $\hat{x}_i = 0$, $i \in N \setminus q$, and the equations in (2.17) then provide initial values for x_D . It is a consequence of the procedure for pivoting on $\hat{a}_{qp}$ that these initial values are negative if the corresponding variable is blocked at level 1.

The method now proceeds to solve (2.17) and there are three possible outcomes.

- A feasible point of (2.17) is obtained. In this case a feasible ascent direction at level 1 is obtained and the degeneracy block has been removed.

- It is found that (2.17) has no feasible solution. It follows that $\hat{x}$ is optimal at level 1.

In both of these cases the simplex method can be returned to standard form by pivoting δ_q back into B , as illustrated in the example, although there is a more efficient alternative that is outlined below. The third outcome is the following.

- A degeneracy block occurs while solving (2.17). In this case the same technique can be used recursively to resolve the situation.

Thus the problem based on (2.2) is regarded as the level 1 problem, problem (2.17) is the level 2 problem, and a degeneracy block while solving (2.17) would give rise to a level 3 problem, and so on. Pivots made at level 2 (or higher levels) are also made in the level 1 system, so that all levels share the same matrix $\hat{A}$ (or a part of it).

In fact there is a more efficient way of implementing the process. Once the level 2 problem has been solved, a return to level 1 can be made while δ_q is nonbasic. Then δ_q becomes the variable to increase at level 1. There are two cases to consider. If $q \neq 0$, and δ_q can be driven up to $-\hat{x}_q$ (equivalently, x_q can be driven up to zero) without any new infeasibilities occurring in x_B , then δ_q is replaced by x_q as a nonbasic variable and the level 1 right-hand side is updated, but no pivot in $\hat{A}$ is required. In this case two pivot steps are saved. Alternatively, if a nondegenerate basic variable, x'_p , say, is reduced to zero, then a pivot on $\hat{a}_{p'q}$ is made, δ_q becomes basic and the level 1 right-hand side is updated. Also δ_q is replaced by x_q and the old value of $\hat{x}_q$ is added to the right-hand side. In this case one pivot is saved.

This case can also be illustrated by Beale's example. Proceeding from (2.12), the nonbasic variable δ_0 is increased. The variable x_7 is driven down to zero, and the resulting pivot on the (x_7, δ_0) element of $\hat{A}$ gives rise to the solution as in (2.14).

A further simplification can also be made. It may be while solving the level L problem ($L > 1$) that optimality in some row (p'' , say) is detected. If this happens, a pivot can be made on element $\hat{a}_{p''q}$, where q corresponds to the nonbasic variable δ_q derived from the target variable at level $L - 1$. It follows from the pivoting rules that optimality of the target variable at level $L - 1$ is detected. By induction it follows that level 1 is optimal. In fact the variables δ_q at levels $L, L - 1, \dots, 2$ swap in and then out of the basis during these interchanges. Thus it is possible to achieve the same outcome more readily, simply by pivoting on the element in row p'' and the column corresponding to the target variable from level 1. This returns to the level 1 problem with just one pivot.

One might ask if it is possible to use other methods for solving the recursively generated feasibility problems such as (2.17). However, if, say, some l_1 sum of infeasibilities is minimized, setting up the relevant objective function at level 2 or above requires an extra vector of storage at each level. If the one-at-a-time method is used, the method may be implemented so that only one scalar is stacked at each level of recursion.

It is possible to give an upper bound on the number of levels of recursion. When a degeneracy block occurs at level L , then the target variable δ_q at that level becomes nonbasic and is set to 1; thus it does not participate in the level $L + 1$ problem. Also there is at least one row with a positive right-hand side in the level L problem (for $L > 1$) that cannot participate in further recursion. Consequently it can be shown that the maximum number of levels is equal to the minimum dimension of the matrix $[\hat{A}|\hat{x}_B]$ (including the x_0 row if in phase II).

An advantage of a method based on the recursive solution of feasibility problems is that it is possible to give a guarantee of termination, both for exact and inexact arithmetic. At any level of recursion, the one-at-a-time method exhibits a *monotonic improvement property*; that is, either the number of infeasible residuals is reducing or the current target function is strictly increasing. The occurrence of degeneracy does not contradict this observation because the solution of the higher level problem either provides a feasible ascent direction for the current level or indicates that the current level is optimal. As in the previous paragraph, the number of levels of recursion is finite. For exact arithmetic we also need to argue that there are finitely many vertices at any level. The above arguments then provide the basis of an inductive proof of termination (see [5], [6]).

For inexact arithmetic a key feature is to ensure that the *stored value* of the target variable x_q (at any level) is strictly improving. If not, then the situation is treated as a degeneracy block, and if necessary, the current stored information is adjusted so as to be consistent with this. Assuming that overflow does not occur, it is then possible, with some care, to ensure that the monotonic improvement property holds for inexact arithmetic. Because the range of floating-point numbers is finite, it can be deduced (without recourse to the finite number of vertices argument, which is not valid for inexact arithmetic) that the algorithm terminates with stored values that indicate a solution (or infeasibility or unboundedness).

In practice, the method would not be implemented as a tableau method (that is, by storing and updating $\hat{A}$). Rather, selected rows and columns of $\hat{A}$ would be obtained as required by making solves with an invertible representation of A_B , which is updated when a pivot operation is made. In phase I of the above method, this is quite straightforward to implement, at any level. In phase II, there is the complication of the added variable x_0 . At level 1, the basis matrix for the extended system is

$$(2.18) \quad \begin{bmatrix} 1 & c_B^T \\ 0 & A_B \end{bmatrix}$$

and the calculation of row 0 of $\hat{A}$ is equivalent to the standard simplex calculation of reduced costs. However, at levels greater than 1, the unit column in (2.18) is replaced by the vector $(c_p, a_p^T)^T$. Then the question of how best to implement solves with this matrix needs to be addressed (although there are obvious ways to proceed). For this and other reasons, the idea has been implemented in an alternative way based on an ASM formulation, in which the need for an extended system does not arise. This is described in section 4.

3. Comparison with other recursive methods. This section describes two alternative recursive methods for resolving degeneracy and considers their relative merits and demerits when compared with the new method of section 2. All these methods share an important advantage that current values of variables at higher levels can be stored by overwriting zero (degenerate) values at lower levels. Consequently very little extra storage is required to implement the recursion. Thus the methods are all serious contenders for use in a practical LP code.

The author has been interested in a particular primal-dual method [5] for resolving degeneracy in LP, which is related to a method of Balinski and Gomory [1]. The method in [5] is also based on the recursive solution of feasibility problems, but these are generated in a different way from the new method. However, the method allows the same type of termination guarantee for exact and inexact arithmetic to be proved,

as for the new method. The method in [5] also depends strongly on the use of the one-at-a-time method.

Consider the situation in which a (level 1) simplex iteration is blocked because of degeneracy; that is, the minimum ratio in (2.4) is $\alpha_{q'p} = 0$. The method in [5] proceeds as follows. As before, all the nondegenerate rows in (2.3) are ignored, and the remaining localized problem,

$$(3.1) \quad \begin{aligned} & \text{maximize} && x_q = \hat{x}_q - \hat{a}_{q\bullet} x_N \\ & \text{subject to} && x_D = -\hat{A}_D x_N \geq 0, \quad x_N \geq 0, \end{aligned}$$

is considered. Problem (3.1) is also degenerate, but its dual is the (level 2) feasibility problem

$$(3.2) \quad \lambda_N = \hat{a}_{q\bullet}^T + \hat{A}_D^T \lambda_D, \quad \lambda_N \geq 0, \quad \lambda_D \geq 0.$$

This system has essentially the same structure as (2.2) and (2.3) and can likewise be solved by the one-at-a-time method. The pivots used in solving (3.2) are also applied to the primal system (2.3). There are three possible outcomes when solving this dual system.

- A feasible point of the dual problem (3.2) is obtained: in this case the elements of $\hat{a}_{q\bullet}$ have become nonnegative and the current solution of the level 1 problem is optimal (an infeasible LP is indicated if in phase I).

- The dual is infeasible: in the one-at-a-time method this is discovered by finding a row of $\hat{A}^T$ in (3.2), row p , say, whose elements are all nonpositive. It then follows in (3.1) that the degeneracy block has been removed (see below for an illustration).

- A degeneracy block occurs in the dual: in this case the dualization process is repeated, giving a level 3 feasibility problem that is resolved in an analogous way.

In Beale's example, after deleting the nondegenerate x_7 row in (2.7), the realization of the level 2 feasibility problem is

$$(3.3) \quad \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{pmatrix} = \begin{pmatrix} -\frac{3}{4} \\ 20 \\ -\frac{1}{2} \\ 6 \end{pmatrix} - \begin{bmatrix} -\frac{1}{4} & -\frac{1}{2} \\ 8 & 12 \\ 1 & \frac{1}{2} \\ -9 & -3 \end{bmatrix} \begin{pmatrix} \lambda_5 \\ \lambda_6 \end{pmatrix}, \quad \lambda_i \geq 0, \quad i = 1, 2, \dots, 6.$$

Note that the matrix in (3.3) is $-\hat{A}_D^T$ for compatibility with (2.3). Clearly a consequence of (2.7) being nonoptimal is that the current right-hand side in (3.3) is infeasible. Taking λ_1 as the target variable, the optimality test chooses λ_6 to increase. The ratio test indicates that no new infeasibilities are introduced and λ_1 can be driven up to zero. A pivot is taken on the (λ_1, λ_6) element in (3.3), giving the system

$$(3.4) \quad \begin{pmatrix} \lambda_6 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{pmatrix} = \begin{pmatrix} \frac{3}{2} \\ 2 \\ -\frac{5}{4} \\ \frac{21}{2} \end{pmatrix} - \begin{bmatrix} \frac{1}{2} & -2 \\ 2 & 24 \\ \frac{3}{4} & 1 \\ -\frac{15}{2} & -6 \end{bmatrix} \begin{pmatrix} \lambda_5 \\ \lambda_1 \end{pmatrix}.$$

Now λ_3 is chosen as the target variable, but the optimality test indicates that it cannot be increased, so the level 2 problem is infeasible. The corresponding level 1

system is now

$$(3.5) \quad \begin{pmatrix} x_0 \\ x_5 \\ x_1 \\ x_7 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} - \begin{bmatrix} \frac{3}{2} & 2 & -\frac{5}{4} & \frac{21}{2} \\ -\frac{1}{2} & -2 & -\frac{3}{4} & \frac{15}{2} \\ 2 & -24 & -1 & 6 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} x_6 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix},$$

and a consequence of the optimality of $\hat{\lambda}_3$ in (3.4) is that there is no longer a degeneracy block for increasing x_3 in (3.5). The next pivot on $\hat{a}_{73}$ then gives the system (2.14), which solves Beale's problem.

The ASM form of this method has been extended to solve QP problems in [6], and has been implemented in the LP/QP code **bqpd**, which is available on request from the author. The same issues apply in regard to termination, particularly in regard to inexact arithmetic, as discussed in the previous section. In practice there is considerable flexibility in the choice of heuristics that can be applied to adjust the stored information when degeneracy is detected. These ideas have been explored in [7] and [6]. As in other LP codes, attention to numerical stability is still important, for example, the avoidance of small pivots that might lead to an ill-conditioned basis. It is felt that the existence of this guarantee of termination for inexact arithmetic, together with the use of other heuristics as in [7] and [6], has contributed strongly to the robustness of **bqpd**.

A feature of **bqpd** is that the recursively generated problems alternate between the primal and dual spaces of the LP. If the steepest-edge criterion were to be used on every iteration, steepest-edge coefficients would be needed in both spaces, which would considerably increase the effort required to update these coefficients. The alternative of using the steepest-edge criterion in **bqpd** only at odd numbered levels is likely to be more efficient but does not take full advantage of the steepest-edge approach. The method of section 2 works in the primal space at all levels, so that the steepest-edge criterion can be used on every iteration, and only primal steepest-edge coefficients need to be maintained.

Another technique for resolving degeneracy is that of Wolfe [13]. As in the new method, the recursively derived problems are in the primal LP variables, so that the technique is also able to make full use of primal steepest-edge coefficients. Wolfe's technique is based on altering degenerate constraints to be strictly feasible when a degeneracy block occurs. For example, in (2.7) the level 2 problem is obtained by ignoring the nondegenerate x_7 row and altering the right-hand sides $\hat{x}_5$ and $\hat{x}_6$ to have some arbitrary positive value. If the value of 1 is chosen, the level 2 problem is to maximize x_0 subject to

$$(3.6) \quad \begin{pmatrix} x_0 \\ x_5 \\ x_6 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} - \begin{bmatrix} -\frac{3}{4} & 20 & -\frac{1}{2} & 6 \\ \frac{1}{4} & -8 & -1 & 9 \\ \frac{1}{2} & -12 & -\frac{1}{2} & 3 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}, \quad x_i \geq 0, \quad i = 1, \dots, 6,$$

and is nondegenerate. The ratio test determines a pivot on $\hat{a}_{61}$, leading to the system

$$(3.7) \quad \begin{pmatrix} x_0 \\ x_5 \\ x_1 \end{pmatrix} = \begin{pmatrix} \frac{3}{2} \\ \frac{1}{2} \\ 1 \end{pmatrix} - \begin{bmatrix} \frac{3}{2} & 2 & -\frac{5}{4} & \frac{21}{2} \\ -\frac{1}{2} & -2 & -\frac{3}{4} & \frac{15}{2} \\ 2 & -24 & -1 & 6 \end{bmatrix} \begin{pmatrix} x_6 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}, \quad x_i \geq 0, \quad i = 1, \dots, 6.$$

Now the optimality test indicates that x_3 should be increased. As all the entries in the x_3 column of $\hat{A}$ are nonpositive, the level 2 problem is seen to be unbounded. Making the same pivot in the level 1 system (2.7) gives rise to (3.5), in which the degeneracy block has been removed.

This example illustrates that in Wolfe's technique a return from level 2 to level 1 is made when *unboundedness* is recognized in the level 2 problem. Heuristics could be used to ensure monotonic improvement of the objective function during the solution of the level 2 problem, but these would *not* imply a guarantee of termination for inexact arithmetic. Indeed if one were allowed to introduce arbitrary errors into a column at any stage, it would be possible to ensure that some positive entries are always present in a column with a negative cost. This would ensure that the level 2 problem never terminates. Although unlikely, it is not impossible to imagine this situation occurring in practice, particularly for highly degenerate large problems for which insufficiently large tolerances have been set.

One referee correctly observes that in all these methods, a return to level 1 is taken when a certain vector is nonpositive (or nonnegative). In Wolfe's method this occurs when the denominators in the ratio test at level 2 are nonpositive. However, there is no mechanism in Wolfe's method that forces these denominators to become nonpositive for inexact arithmetic. In the methods based on solving feasibility problems at level 2, the stored value of the right-hand side vector improves monotonically on every iteration and must ultimately become feasible if the method stays at level 2 or above. Thus a return to level 1 must occur in a finite number of iterations.

4. Implementation as an ASM. In this section, the general approach to solving an LP problem by an ASM is outlined. The new method for resolving degeneracy is explained in an ASM context, and the computational steps are derived by relating the ASM to the method described in section 2. Details of implementation of the new degeneracy method are also discussed and it is shown that the new method is just as convenient in this respect as the method in [5]. Some numerical experiments on the SOL test problem set suggest that a worthwhile improvement in performance is gained from the use of the steepest-edge criterion as against the Dantzig criterion, in an ASM formulation.

In contrast to (2.1), the standard form of an LP problem for an ASM is regarded as being

$$(4.1) \quad \begin{array}{ll} \text{minimize} & f = c^T x, \quad x \in \mathbb{R}^n, \\ \text{subject to} & A^T x \geq b, \end{array}$$

where $A \in \mathbb{R}^{n \times m}$ and $b \in \mathbb{R}^m$, with $m \geq n$. Any simple bounds on the variables are subsumed into the constraints $A^T x \geq b$. Equality constraints are not specifically considered, although it is a simple matter to modify the ASM to do this. If the bounds $x \geq 0$ are included amongst the constraints in (4.1), it is possible to transform from (4.1) to (2.1) by using slack variables, in which case the standard primal simplex method and the ASM (with Dantzig pricing) follow the same path to the solution (see, for example, [4, p. 161]) and differ only in how the matrix computations are organized. In fact it is common practice in commercial simplex codes to include an auxiliary variable in every constraint row (excluding the simple bounds). These variables may be interpreted as slack, surplus, or artificial variables as appropriate. When the steepest-edge criterion is used, the ASM and simplex methods are no longer equivalent in the above sense because the inclusion of these auxiliary variables changes

the lengths of the edges. Thus the relative performance of these two methods is in general different and is problem dependent.

Another way of using an ASM is to apply it to the dual of (2.1),

$$(4.2) \quad \begin{aligned} & \text{maximize} && b^T \lambda, && \lambda \in \mathbb{R}^m, \\ & \text{subject to} && A^T \lambda \leq c. \end{aligned}$$

This is referred to as the *dual simplex method*. The use of steepest-edge pricing in this context gives rise to the method *Steepest 1* of Forrest and Goldfarb [8]. This is different again from the steepest-edge methods described in the previous paragraph and often shows up well in the experiments described in [8].

A brief description of phase II of the ASM applied to (4.1) is now given (see, for example, [4] for more details). In phase II, the iterates of the ASM are vertices of the feasible region, and each iteration takes the form of a descent step from the current vertex to an adjacent vertex. Let $a_i \in \mathbb{R}^n$ denote column i of A (the normal vector of the i th constraint). The *active set* denoted by $\mathcal{A}$ is a set of n constraint indices that determine the current vertex $\hat{x}$, which may thus be obtained by solving the equations $a_i^T x = b_i$, $i \in \mathcal{A}$. The matrix $A_{\mathcal{A}}$, whose columns are the normal vectors a_i , $i \in \mathcal{A}$, is nonsingular and plays an important part. Also the set $\mathcal{I}$ denotes the complement of $\mathcal{A}$ and contains the indices of the inactive constraints and possibly some degenerate constraints.

If $\hat{x}$ is nondegenerate in the sense that $a_i^T \hat{x} > b_i$ for all $i \in \mathcal{I}$, then there are n feasible edges emanating from $\hat{x}$, denoted by d_i , $i = 1, 2, \dots, n$ (see (4.12) below). It is readily shown that these vectors may be taken to be the columns of the matrix $A_{\mathcal{A}}^{-T}$. The slope of f along any edge d_i is given by $\hat{\lambda}_i = c^T d_i$, which can be interpreted as a Lagrange multiplier. The ASM iteration proceeds as follows. The current vertex $\hat{x}$ is optimal if $\hat{\lambda}_i \geq 0$, $i = 1, 2, \dots, n$. Otherwise an index p is selected such that $\hat{\lambda}_p < 0$, and this determines the edge d_p that is chosen as the search direction. This is essentially the optimality test described in section 2. Next the step along the edge to the nearest vertex is calculated (see the text following (4.16) below). This is essentially the ratio test of section 2. If $q \in \mathcal{I}$ denotes the index of the constraint that becomes active in the search, then q replaces p in $\mathcal{A}$ and a_q replaces a_p in $A_{\mathcal{A}}$. Note that this exchange redefines the vectors d_i and their lengths at the new vertex.

In the traditional approach (that adopted by Dantzig [2] in the simplex method), the choice

$$(4.3) \quad p = \arg \min_i \hat{\lambda}_i$$

is made in the optimality test, with the hope that the most negative slope will provide a significant reduction in f . However, this thinking neglects the fact that the lengths of the d_i may differ. Therefore, in the steepest-edge method the edges are normalized, giving rise to the choice

$$(4.4) \quad p = \arg \min_i \frac{\hat{\lambda}_i}{\|d_i\|_2}.$$

This choice requires that the *steepest-edge coefficients*

$$(4.5) \quad \gamma_i = d_i^T d_i, \quad i = 1, 2, \dots, n,$$

are available. The main contribution of Goldfarb and Reid [9] and Forrest and Goldfarb [8] lies in pointing out that it is possible to update steepest-edge coefficients at

reasonable cost. When these are defined by (4.5), the appropriate formula is

$$(4.6) \quad \gamma_i^\# = \gamma_i - 2 \frac{t_i}{t_p} d_i^T d_p + \left(\frac{t_i}{t_p} \right)^2 \gamma_p, \quad i = 1, 2, \dots, n, \quad i \neq p,$$

where $\#$ denotes successor and $t = A_{\mathcal{A}}^{-1} a_q$. The vector t may be available from the ratio test, or when updating an invertible representation of $A_{\mathcal{A}}$. Equation (4.6) needs the values $d_i^T d_p$, which can be written as $[A_{\mathcal{A}}^{-1} d_p]_i$ by definition of d_i . These values are not available and require an extra solve with $A_{\mathcal{A}}$. Equation (4.6) is given by Forrest and Goldfarb [8] in the context of the dual simplex method.

There are also some issues regarding how the coefficients γ_i are initialized. In a *cold start* to the ASM, the active constraints are just simple bounds, $A_{\mathcal{A}} = I$, and the coefficients γ_i are all initialized to unity. However, in a *warm start*, an arbitrary set of active constraints is prescribed, including for example any equations in the LP problem (4.17). Initializing the coefficients γ_i in this case is an expensive calculation, so that the advantage of steepest-edge methods is more apparent with a cold start.

If the ratio test determines a step of zero length, then a degeneracy block has arisen and must be resolved. As in previous sections, the nondegenerate constraints in $\mathcal{I}$ are ignored, and the localized problem

$$(4.7) \quad \begin{aligned} &\text{minimize} && f = c^T x, && x \in \mathbb{R}^n, \\ &\text{subject to} && a_i^T x \geq 0, && i \in \mathcal{A} \cup D, \end{aligned}$$

is considered, where $D = \{i \in \mathcal{I}, a_i^T \hat{x} = b_i\}$ is the set of degenerate constraints. The new method in this paper is obtained by seeking a feasible descent direction d for this problem, normalized so that $c^T d = -1$. That is to say, a solution of the feasibility problem

$$(4.8) \quad c^T d = -1, \quad A_{\mathcal{A}}^T d \geq 0, \quad A_D^T d \geq 0$$

is sought, where A_D denotes the matrix with columns a_i , $i \in D$. This problem is the analogue of (2.17) above. The outcome is the same, namely, that either a feasible descent direction for (4.7) is determined, or it is established that none exists, in which case $\hat{x}$ is optimal. The precise details for solving (4.8) are most readily explained by expressing the ASM in an analogous form to (2.2), and this is done below. Roughly speaking, the constraint $c^T d = -1$ is made active, with c replacing the constraint normal a_p in $A_{\mathcal{A}}$ for some p . This is equivalent to the initial pivot on $\hat{a}_{qp}$ in the form of the method described in section 2. Then the one-at-a-time method is used to drive the infeasible constraint residuals in (4.8) up to zero. The vector c is removed from $A_{\mathcal{A}}$ once the degeneracy block is resolved.

To define the method more precisely, the ASM is expressed in a form analogous to (2.2) and (2.3) so that the steps of the method as described in section 2 can be followed. For an ASM, the constraints in (4.1) may be expressed in terms of a vector of *residuals*

$$(4.9) \quad r = A^T x - b \geq 0,$$

which may also be regarded as slack variables. On each iteration of the ASM the sets $\mathcal{A}$ and $\mathcal{I}$ are given and a permutation matrix P is introduced such that

$$(4.10) \quad Pr = \begin{bmatrix} r_{\mathcal{A}} \\ r_{\mathcal{I}} \end{bmatrix} = \begin{bmatrix} A_{\mathcal{A}}^T \\ A_{\mathcal{I}}^T \end{bmatrix} x - \begin{pmatrix} b_{\mathcal{A}} \\ b_{\mathcal{I}} \end{pmatrix}.$$

Here $A_{\mathcal{I}}$ is the matrix whose columns are the normal vectors a_i , $i \in \mathcal{I}$, and vectors such as $r_{\mathcal{A}}$ collect the elements r_i , $i \in \mathcal{A}$. The current value of the active constraint residuals is $\hat{r}_{\mathcal{A}} = 0$, so (4.10) defines the current vertex to be

$$(4.11) \quad \hat{x} = A_{\mathcal{A}}^{-T} b_{\mathcal{A}}.$$

It is now convenient to regard $r_{\mathcal{A}}$ as the vector of independent variables of the problem, rather than x . Thus x is eliminated from the problem using

$$(4.12) \quad x = A_{\mathcal{A}}^{-T}(b_{\mathcal{A}} + r_{\mathcal{A}}) = \hat{x} + A_{\mathcal{A}}^{-T} r_{\mathcal{A}}.$$

Observe in passing that (4.12) shows that the edges emanating from $\hat{x}$ are the columns of $A_{\mathcal{A}}^{-T}$ (because an edge is obtained by relaxing one active constraint residual while keeping the others at zero). It follows from (4.10) and (4.12) that the residuals of constraints in $\mathcal{I}$ can be written as

$$(4.13) \quad \begin{aligned} r_{\mathcal{I}} &= A_{\mathcal{I}}^T(\hat{x} + A_{\mathcal{A}}^{-T} r_{\mathcal{A}}) - b_{\mathcal{I}} \\ &= \hat{r}_{\mathcal{I}} - \hat{A} r_{\mathcal{A}}, \end{aligned}$$

where $\hat{r}_{\mathcal{I}} = A_{\mathcal{I}}^T \hat{x} - b_{\mathcal{I}}$ denotes the current value of residuals in $\mathcal{I}$, and

$$(4.14) \quad \hat{A} = -A_{\mathcal{I}}^T A_{\mathcal{A}}^{-T}.$$

Likewise, the objective function can be expressed as

$$(4.15) \quad f = c^T x = c^T(\hat{x} + A_{\mathcal{A}}^{-T} r_{\mathcal{A}}) = \hat{f} + \hat{\lambda}^T r_{\mathcal{A}},$$

where $\hat{f}$ is the current function value and $\hat{\lambda} = A_{\mathcal{A}}^{-1} c$ is the vector of Lagrange multipliers. Equations (4.13) and (4.15), together with the bounds (4.9), can be written as the system

$$(4.16) \quad \begin{aligned} r_0 &= -\hat{f} - \hat{\lambda}^T r_{\mathcal{A}}, \\ r_{\mathcal{I}} &= \hat{r}_{\mathcal{I}} - \hat{A} r_{\mathcal{A}}, \quad r_{\mathcal{I}} \geq 0, \quad r_{\mathcal{A}} \geq 0, \end{aligned}$$

where $r_0 = -f$. This is equivalent to the standard simplex system derived from (2.2), (2.3), and (2.5), with $r_{\mathcal{I}}$ and $r_{\mathcal{A}}$ being synonymous with x_B and x_N , respectively.

Quantities used in the ASM are obtained from selected rows and columns of $\hat{A}$ in (4.14), although in practice this matrix itself is not stored or updated. In phase I, if r_q is the target residual to be increased, multipliers for the optimality test are obtained from row q of $\hat{A}$ and are given by the row vector $-a_q^T A_{\mathcal{A}}^{-T}$. This requires a solve with $A_{\mathcal{A}}$. In phase II, r_0 is the target variable to increase and $\hat{\lambda}$ is the vector of multipliers. In the optimality test, an active constraint residual r_p is chosen to be increased. However, it is emphasized that the relevant steepest-edge criterion is that obtained by normalizing the edges in x -space, as in (4.4), and not in a space that includes the slack variables r . It follows from (4.12) that the current point x moves from $\hat{x}$ in a direction $d_p = A_{\mathcal{A}}^{-T} e_p$ that defines an edge of the feasible region. The definition $\hat{\lambda} = A_{\mathcal{A}}^{-1} c$ of the Lagrange multipliers, and $\hat{\lambda}_p < 0$, ensures that d_p is a descent direction for f . Denominators for the ratio test (2.4) are obtained from column p of $\hat{A}$ in (4.14), where p is the index chosen in the optimality test. This column is the vector $-A_{\mathcal{I}}^T A_{\mathcal{A}}^{-T} e_p$ and requires a solve with $A_{\mathcal{A}}^T$ on the unit vector e_p , followed by inner products with the columns of $A_{\mathcal{I}}$. The ratio test corresponds

to finding the first constraint boundary, indexed by q' , that is crossed by the edge $\hat{x} + \alpha d_p$, $\alpha > 0$. If this boundary prevents an improvement in f being made, then a degeneracy block has occurred. This can be resolved likewise by following the steps in section 2 applied to the system (4.16). Rows and columns of $\hat{A}$ are obtained by solves with $A_{\mathcal{A}}$ as above. Each pivot operation in the method causes an exchange of indices between $\mathcal{A}$ and $\mathcal{I}$. This causes one column of $A_{\mathcal{A}}$ to be exchanged with a column of $A_{\mathcal{I}}$. To expedite the solves with $A_{\mathcal{A}}$, it is important, if large problems are to be solved efficiently, that a sparse invertible representation of $A_{\mathcal{A}}$ should be available, which can be updated efficiently when columns are exchanged. A good reference to recent developments in this respect is given by Suhl [11].

The new method has been implemented for the more general form of (4.1) given by

$$(4.17) \quad \begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & l \leq \begin{pmatrix} x \\ A^T x \end{pmatrix} \leq u, \end{array}$$

where $A \in \mathbb{R}^{n \times m}$. Many of the features of **bqpd** (see [7], [6]) have been incorporated into the code and similar data structures are used. The new method shares the advantage of that in [5] that values of residuals at higher levels can be stored by overwriting zero (degenerate) residuals at lower levels. Consequently very little extra storage is required to implement the recursion. A vector of residuals r stores (in natural order) the current value of the residual or multiplier for each constraint. This is pointed to by the indexing vector ls (see [6]) that implements the permutation P in (4.10). Thus at level 1, ls contains the indices of $\mathcal{A}$ in the first n positions followed by those of $\mathcal{I}$. Special attention in the algorithm is given to equality constraints ($l_i = u_i$) and to pseudo-bounds (psb's) ($l_i < \hat{x}_i < u_i$) that arise in $\mathcal{A}$. Hence pointers peq and $ppsb$ to the end of the groups of equality and psb constraints in $\mathcal{A}$ are also kept. The length of ls and r is $n+m$ or $n+m+1$ according to whether the algorithm is in phase I or II. A sign is associated with each element of ls , which indicates whether the lower (+) or upper (−) bound is currently being considered. At level L the indices in $\mathcal{I}$ are grouped according to the highest level at which they participate, and there is a list of pointers lp whose elements point to the last entry in each group of indices. The general structure of ls may be pictured as

$$\begin{array}{ccccccc} & & n-L+1 \downarrow & & & & \\ \boxed{\text{equalities}} & \boxed{\text{psb's}} & \boxed{\text{rest of } \mathcal{A}} & \boxed{\text{level } L} & \boxed{\dots} & \boxed{\text{level } 2} & \boxed{\text{level } 1} \\ peq \uparrow & ppsb \uparrow & & lp(L) \uparrow & & lp(2) \uparrow & lp(1) \uparrow \end{array} .$$

The index of the target variable r_q at level j is stored in $ls(lp(j))$. In general the elements of $\mathcal{A}$ are the first $n-L+1$ elements in ls and $ls(lp(j))$ for $j = 2, \dots, L$.

A preliminary LP code to implement the steepest-edge ASM on the primal LP (4.17) has been written and shows considerable promise. Results are given for a variety of smallish LPs in the SOL test set (a subset of the netlib set), the characteristics of which are shown in Table 1. The numbers of pivots required by the new code are shown in Table 2. The method is initialized by taking as many equality constraints as possible into the active set and completing the set with simple bounds chosen to keep $A_{\mathcal{A}}$ well-conditioned. The effect of using the steepest-edge criterion (4.4) as against the Dantzig criterion (4.3) is shown, and an improvement mostly in the range

TABLE 1
Characteristics of SOL test problems.

Test problems	Numbers of		
	Variables n	General constraints	Nonzero elements in A
adlittle	97	56	465
share2b	79	96	730
share1b	225	117	1182
beaconfd	262	173	3476
israel	142	174	2358
brandy	249	220	2150
e226	282	223	2767
capri	353	271	1786
bandm	472	305	2659
stair	467	356	3857
etamacro	688	400	2489

TABLE 2
Comparison of total numbers of pivots required.

Test problems	New method		bqpdc (not steepest-edge)	
	With s.edge	Without s.e.	Dense	Sparse
adlittle	78 (18)*	154 (20)	119 (11)	89 (9)
share2b	87 (51)	125 (88)	110 (66)	90 (55)
share1b	237 (153)	374 (243)	387 (225)	262 (127)
beaconfd	37 (0)	36 (0)	31 (0)	28 (0)
israel	177 (10)	377 (117)	399 (8)	394 (8)
brandy	209 (103)	317 (225)	381 (319)	274 (226)
e226	367 (72)	703 (127)	663 (144)	574 (88)
capri	313 (239)	479 (334)	480 (367)	484 (405)
bandm	305 (160)	588 (272)	522 (282)	446 (211)
stair	341 (207)	515 (318)	636 (342)	592 (337)
etamacro	557 (308)	873 (452)	1293 (829)	947 (530)

* Bracketed figure is number of pivots in phase I.

of 30%~50% is observed. A comparison against the bqpdc code that also uses the Dantzig criterion is given. This is not directly comparable because a different method is used in phase I, but nonetheless a significant improvement is also apparent. The differing results for the dense and sparse matrix algebra packages for bqpdc are caused by the difference between partial pivoting and Markowitz pivoting in the crash start and the random effect of differing round-off propagation. The improvement obtained by steepest-edge is won at the expense of an extra solve with A_A on each iteration and the cost of initializing the γ_i , so a more important statistic is the total computing time required to solve each problem. This is shown in Table 3. In this case, the overall improvement is approximately 20%, although there is considerable variation from problem to problem.

The main conclusion from these results is that use of the steepest-edge criterion gives a worthwhile improvement for an ASM, particularly as there are other benefits from having steepest-edge coefficients available (see the next section). It is also of some interest to ask how the new method might compare with a steepest-edge version

TABLE 3
Comparison of computing time (seconds on a Sun ELC).

Test problems	New method (dense factors)		bqpdp (not steepest-edge)	
	With s.edge	Without s.e.	Dense	Sparse
adlittle	1.5	2.4	7.7	1.6
share2b	1.8	2.0	6.5	1.6
share1b	21.6	27.6	90.4	12.5
beaconfd	11.1	8.9	25.9	3.1
israel	6.7	16.1	58.0	14.6
brandy	42.5	45.8	123.6	20.0
e226	42.4	69.1	279.5	33.7
capri	74.4	97.2	306.2	27.8
bandm	223.1	247.1	504.8	46.9
stair	192.1	232.8	639.5	163.2
etamacro	379.9	406.9	2461.1	70.8

of **bqpdp**, were one available. It is unlikely to be cost effective to do this by maintaining steepest-edge coefficients in both primal and dual spaces, which requires two extra solves per iteration. However, there is also the possibility of implementing **bqpdp** with steepest-edge selection only at the odd levels of recursion. To estimate the likely outcome, the proportion of iterations taking place at odd and even levels for the dense version of **bqpdp** have been counted. For some problems, almost all the iterations take place at level 1 and the steepest-edge **bqpdp** code would behave almost identically to the new code. On the other hand, for the etamacro problem, 52% of iterations are at even levels, suggesting a potential for greater improvement with the new method. On all problems excluding etamacro, 11% of iterations of **bqpdp** are at even levels. Set against this is the fact that the new method may take an extra pivot due to the target function being pivoted into the basis matrix. Overall it is unlikely that the new method would show very much improvement over **bqpdp** with steepest-edge selection at odd levels. However, the new method does provide a viable alternative to the method used in **bqpdp** and might therefore be preferred for organizational reasons, particularly in that it avoids the primal-dual alternation of **bqpdp**.

The opportunity has also been taken to improve the efficiency of the dense matrix package used in the new code. This gives considerably improved performance over the dense code used by **bqpdp**, as Table 3 indicates. However, the new code is not finalized, and decisions have yet to be made about how best to truncate near-zero quantities that arise due to rounding errors. There are also some options in regard to how phase I is carried out. At levels $L > 1$, the one-at-a-time method makes it easy to stack up the cost functions at different levels of recursion. However, the code at level 1 is not used at higher levels since it caters to various special features such as two-sided bounds, equalities, psb's, etc. Therefore it is straightforward to use a different phase I method at level 1, for example, by minimizing an l_1 sum of constraint violations in some way.

The motivation for using a steepest-edge criterion is based on an implicit assumption that the variables are well-scaled in some sense. The results above are obtained without using any scaling. An attempt was made to estimate scaling factors for the variables of the SOL test problems so that the effect of scaling could be assessed. It was found that an even greater improvement was obtained by using the steepest-edge criterion, as against the Dantzig criterion, on these scaled problems.

5. Steepest-edge and condition estimates. In this and the next section it is shown how the availability of steepest-edge coefficients provides useful spin-off in that it enables realistic estimates of the actual condition of the systems that arise in an ASM to be assessed. The estimates arise from the idea of *expected condition* introduced in [3]. This idea is summarized below, and the arguments that support its effectiveness are reviewed. Application of the idea in an ASM context is described, and it is shown that availability of steepest-edge coefficients makes the estimates readily available. A table of condition estimates for the SOL problems is given and provokes some interesting questions.

First, the idea of expected conditioning is reviewed. Consider the nonsingular system

$$(5.1) \quad Ax = b,$$

and let a perturbation E in A induce a perturbation h in x , so that

$$(5.2) \quad (A + E)(x + h) = b.$$

It is assumed that $A + E$ is also nonsingular so that

$$(5.3) \quad h = -(A + E)^{-1}Ex$$

relates h to E . In practice E is not known, but some estimate of its magnitude is often available. Condition estimates represent an attempt to express the magnitude of h in terms of the magnitude of E , usually in a relative sense.

A common technique is to take a norm inequality in (5.3) and rearrange, giving

$$(5.4) \quad \frac{\|h\|}{\|x\|} \leq \kappa_{\text{norm}} \frac{\|E\|}{\|A\|},$$

where

$$(5.5) \quad \kappa_{\text{norm}} = \|(A + E)^{-1}\|\|A\|$$

is the *norm condition number*. (Various other ways of presenting this result are possible.) The ratio $\|E\|/\|A\|$ is regarded as a measure of the relative error of the matrix A , and (5.4) indicates that the relative error in the solution is at most κ_{norm} times the relative error in the matrix. Often κ_{norm} is used as an estimate of the actual condition of (5.1) in some sense. However, it is argued below that this can be unreliable.

In the expected conditioning approach, a particular assumption is made about how the perturbation E arises, which enables a statistical estimate of its effect to be made. It is most convenient to assume that E is defined by

$$(5.6) \quad e_{ij} = \varepsilon_{ij}a_{ij} \quad \forall i, j,$$

where the ε_{ij} are sampled from a uniform distribution in $[-\varepsilon, \varepsilon]$ for some ε . Thus ε measures the componentwise relative error in A . The ε_{ij} have mean and variance given by

$$(5.7) \quad \mathcal{E}(\varepsilon_{ij}) = 0, \quad \mathcal{E}(\varepsilon_{ij}^2) = \frac{1}{3}\varepsilon^2,$$

where $\mathcal{E}(\cdot)$ denotes expected value, and are assumed to be statistically independent, that is

$$(5.8) \quad \mathcal{E}(\varepsilon_{ij}\varepsilon_{kl}) = 0 \quad \forall i, j \neq k, l.$$

It is also assumed that the effect of variations in E on the value of $(A + E)^{-1}$ can be ignored, so that $(A + E)^{-1}$ can be approximated by A^{-1} . It is convenient to introduce the notation that a vector or matrix in *square* brackets denotes the corresponding vector or matrix in which all the elements have been *squared*. It then follows from (5.3), (5.6), (5.7), and (5.8) that

$$(5.9) \quad \mathcal{E}([h]) \simeq \frac{1}{3}\varepsilon^2[A^{-1}][A][x].$$

Since $\|h\|^2 = e^T[h]$ (where $e^T = (1, 1, \dots, 1)$ and $\|\cdot\|$ is the 2-norm), it follows that

$$(5.10) \quad \begin{aligned} \mathcal{E}\left(\frac{\|h\|^2}{\varepsilon^2\|x\|^2}\right) &\simeq \frac{1}{3}e^T[A^{-1}][A][x]/e^T[x] \\ &\simeq \frac{1}{3}\theta^T[x]/e^T[x], \end{aligned}$$

where $\theta^T = e^T[A^{-1}][A]$. This gives the expected value of the square of the actual condition of the problem for a particular x . It is argued in [3] that the estimate

$$(5.11) \quad \mathcal{E}\left(\frac{\|h\|}{\varepsilon\|x\|}\right) \simeq 0.8\sqrt{\mathcal{E}\left(\frac{\|h\|^2}{\varepsilon^2\|x\|^2}\right)} \simeq 0.462\sqrt{(\theta^T[x]/e^T[x])}$$

is adequate for practical purposes. This provides an estimate

$$(5.12) \quad \kappa_x = 0.462\sqrt{(\theta^T[x]/e^T[x])}$$

of the actual condition of a particular system $Ax = b$. An attainable upper bound over all $x \neq 0$ is readily deduced as

$$(5.13) \quad \kappa_A = 0.462\sqrt{\|\theta\|_\infty} \simeq \max_{x \neq 0} \mathcal{E}\left(\frac{\|h\|}{\varepsilon\|x\|}\right),$$

which may be regarded as defining a condition number κ_A that is relevant to all systems involving A . It is important to realize that (5.13) has been obtained by *averaging* over E and *bounding* over x , in contrast to (5.5). It is also possible (see [3]) to obtain lower bounds on the expected condition number in a similar way.

Practical experience (see [3]) indicates that (5.12) and (5.13) provide much more accurate estimates of actual condition than does (5.5). An important property of (5.12) and (5.13) is that they are invariant under row scaling of the equations (5.1). Moreover, κ_A and κ_{A^T} differ and so can reveal differences in condition between $Ax = b$ and $A^Ty = c$. The matrix

$$(5.14) \quad L = \begin{bmatrix} 0.9525 & & & & \\ 0.6350 & 0.2245 & & & \\ 0.4762 & 0.2694 & 0.05499 & & \\ 0.3810 & 0.2694 & 0.09427 & 0.01361 & \\ 0.3175 & 0.2566 & 0.11780 & 0.03024 & 0.003381 \end{bmatrix},$$

due to Wilkinson [12, p. 234] and rounded to four decimal places, provides an example of this for which $\kappa_L = 174$ and $\kappa_{L^T} = 3.12$. Taking the right-hand side as e and solving

$Lx = e$ and $L^T y = e$ give $\kappa_x = 62.2$ and $\kappa_y = 1.87$. In practice we would like to regard the true value of actual condition as being $\|h\|/(u\|x\|)$, where u is the unit round-off. Solving the systems in single precision on a Sun ELC, taking $u = 6.10^{-8}$, and estimating h using iterative refinement give the actual condition of $Lx = e$ as 180 and that of $L^T x = e$ as 0.973. Thus the expected condition estimates predict the actual condition within a factor of about 3 for these systems and show up the considerable difference in the condition of L and L^T . Other experiments reported in [3] also show good agreement. In contrast, the norm condition number for both L and L^T is 1239 for the 2-norm and larger for other common norms. This gives a very pessimistic estimate of actual condition and does not predict the difference in condition between L and L^T . It is also of interest that the LINPACK condition estimator in Matlab gives condition estimates of 1554 for $Lx = b$ and 1303 for $L^T x = b$ so it again does not provide adequate estimates of actual condition.

The main purpose of this section is to relate these ideas to the steepest-edge coefficients γ_i defined in (4.5). In an ASM, the current vertex is defined by the system

$$(5.15) \quad A_{\mathcal{A}}^T \hat{x} = b_{\mathcal{A}}.$$

The appropriate value of θ for calculating $\kappa_{\hat{x}}$ and $\kappa_{A_{\mathcal{A}}^T}$ (see (5.12) and (5.13)) is

$$(5.16) \quad \theta^T = e^T [A_{\mathcal{A}}^{-T}] [A_{\mathcal{A}}^T].$$

Since d_i is a column of $A_{\mathcal{A}}^{-T}$, it follows that $[d_i] = [A_{\mathcal{A}}^{-T}] e_i$ and hence

$$(5.17) \quad \gamma_i = d_i^T d_i = e^T [d_i] = e^T [A_{\mathcal{A}}^{-T}] e_i.$$

Thus the coefficients θ required in (5.12) and (5.13) are given by

$$(5.18) \quad \theta = [A_{\mathcal{A}}] \gamma.$$

The availability of steepest-edge coefficients γ thus enables the vector θ , and hence the condition estimates (5.12) and (5.13), to be readily computed, particularly when $A_{\mathcal{A}}$ is a sparse matrix. There are various ways in which these estimates might be used, related to the detection of ill-conditioning and the setting of suitable tolerances. It is advantageous that (5.18) is invariant under scaling of the constraints, so that (5.12) and (5.13) provide a more reliable indication of condition than might be obtained, say, by examining $\|\gamma\|$. However, because the 2-norm is used to measure x in (5.10) and (5.11), it is still important to scale the components of x to have similar magnitude if at all possible.

In practice the method of solving (5.15) or related systems takes account of unit columns in $\mathcal{A}$ in such a way that no rounding error is attributed to these columns. Thus it is appropriate to modify the assumption about E so that no perturbation of the unit columns is made. The appropriate change to (5.18) is to replace any unit columns of $[A_{\mathcal{A}}]$ by zero columns. This modified form has been used in all calculations.

If steepest-edge coefficients for the dual system $A_{\mathcal{A}} \lambda = c$ are available, then these can similarly be used to determine condition numbers κ_{λ} and $\kappa_{A_{\mathcal{A}}}$ that predict the condition of the dual system. However, with the new method, these estimates are not readily available. Some earlier calculations for the SOL test set, in which both primal and dual steepest-edge coefficients were available, indicated that in most cases the condition numbers κ_x and κ_{λ} were of similar magnitude.

TABLE 4
Expected condition estimates for SOL test problems.

Test problems	Final active set		Max over all iterations	
	$\kappa_{\hat{x}}$	$\kappa_{A_{\mathcal{A}}^T}$	κ_{d_p}	$\kappa_{A_{\mathcal{A}}^T}$
adlittle	5.08	37.9	16.9	432
share2b	34.4	168	463	64108
share1b	18.5	3391	172	77829
beaconfd	0.541	3.82	657	26534
israel	1.65	653	163	7558
brandy	1.26	148	3522	23342
e226	2.03	1604	86.6	13536
capri	2.05	686	110	21945
bandm	0.953	92.7	121	17258
stair	2.16	236	14.7	51592
etamacro	0.944	222	146	741215

Expected condition estimates obtained while solving the SOL test problems are given in Table 4. For the final active set, the numbers $\kappa_{\hat{x}}$ and $\kappa_{A_{\mathcal{A}}^T}$ are given. In fact, the current vertex $\hat{x}$ is not obtained by solving (5.15) but by updating the previous estimate with a multiple of the vector d_p , which is determined by solving the system $A_{\mathcal{A}}^T d_p = e_p$. Thus the value of κ_{d_p} is more relevant to the calculation that is actually made on an intermediate iteration. Therefore the maximum values of κ_{d_p} and $\kappa_{A_{\mathcal{A}}^T}$ over all iterations are also tabulated. As expected, the numbers $\kappa_{\hat{x}}$ and κ_{d_p} are seen to provide much sharper estimates of actual condition than $\kappa_{A_{\mathcal{A}}^T}$. However, it may be a useful rule of thumb that the latter is about a factor of 100 larger.

These experiments provoke interesting questions regarding how much precision is needed to solve LP problems by simplex-like methods. For the final active set, the systems here are remarkably well-conditioned. Even the maximum value of κ_{d_p} over all iterations is quite small, the worst case being the brandy problem. With this possible exception, one might therefore conclude that four decimal digit floating-point arithmetic should be sufficient to provide at least one or two digits of accuracy in the final solution. In fact most LP codes use double precision, and I suspect that they might have the occasional difficulty with the above set of problems in single precision, and even more so in four digit arithmetic. I feel that we still have a lot to learn about the interaction of precision and condition in simplex and ASM solvers, which could have important implications for the reliability of such codes.

6. Conditioning of the null space matrix. In quadratic programming and nonlinear programming by active set methods, an important concept is that of the null space matrix Z whose columns provide a basis for directions within the active constraint manifold. A general formulation (e.g., [4]) is to write

$$(6.1) \quad B = [A_{\mathcal{A}} \quad V], \quad B^{-1} = \begin{bmatrix} Y^T \\ Z^T \end{bmatrix},$$

where $A_{\mathcal{A}} \in \mathbb{R}^{n \times m}$ ($m \leq n$) is the matrix whose columns are the normal vectors of the constraints in the active set, and V is some convenient matrix that is adjoined to make B nonsingular. Null space methods require the ability to calculate vectors of the form Zu and $Z^T u$. This is done implicitly; for example, to calculate the vector

$x = Zu$, the system

$$(6.2) \quad B^T x = \begin{pmatrix} 0 \\ u \end{pmatrix}$$

is solved.

If the primal steepest-edge coefficients are available, they measure the lengths (squared) of the columns of Y and Z . However, it is the case, as in section 5, that the condition of Zu and $Z^T u$ is in general different. Primal steepest-edge coefficients only enable us to say something about Zu , for which κ_{B^T} applies, by virtue of (6.2) and (5.15) ff. in the previous section.

A particularly important case is when V is a selection of columns from the unit matrix. In this case we can assume that these columns do not contribute to the perturbation matrix E . Then (5.13) and (5.18) simplify to give

$$(6.3) \quad \theta = [A_{\mathcal{A}}]\gamma_{\mathcal{A}}, \quad \kappa_{B^T} = 0.462\sqrt{\|\theta\|_{\infty}},$$

where $\gamma_{\mathcal{A}}$ collects the elements of γ that correspond to indices in $\mathcal{A}$. Of course (6.3) may be further simplified by zeroing any unit columns in $[A_{\mathcal{A}}]$ when computing θ , for the same reasons. Again (6.3) is readily computed and provides a measure of possible ill-conditioning in B^T , which could for example indicate that a change in the definition of V is needed. Apropos to this situation, it may become necessary to make pivoting tests based on the magnitude of $z_i^T u$ for some u , where z_i denotes a column of Z . The availability of steepest-edge coefficients suggests that such tests should be normalized by comparing the magnitudes of $(z_i^T u)^2/\gamma_i$.

REFERENCES

- [1] M. L. BALINSKI AND R. E. GOMORY, *A mutual primal-dual simplex method*, in Recent Advances in Mathematical Programming, R. L. Graves and P. Wolfe, eds., McGraw-Hill, New York, 1963, pp. 17–28.
- [2] G. B. DANTZIG, *Linear Programming and Extensions*, Princeton University Press, Princeton, NJ, 1963.
- [3] R. FLETCHER, *Expected conditioning*, IMA J. Numer. Anal., 5 (1985), pp. 247–273.
- [4] R. FLETCHER, *Practical Methods of Optimization*, 2nd ed., John Wiley, Chichester, 1987.
- [5] R. FLETCHER, *Degeneracy in the presence of roundoff errors*, Linear Algebra Appl., 106 (1988), pp. 149–183.
- [6] R. FLETCHER, *Resolving degeneracy in quadratic programming*, Ann. Oper. Res., 47 (1993), pp. 307–334.
- [7] R. FLETCHER AND J. A. J. HALL, *Towards reliable linear programming*, in Numerical Analysis 1989, Pitman Res. Notes Math. Ser. 228, D. F. Griffiths and G. A. Watson, eds., Longman, Harlow, UK, 1990, pp. 89–104.
- [8] J. J. FORREST AND D. GOLDFARB, *Steepest-edge simplex algorithms for linear programming*, Math. Programming, 57 (1992), pp. 341–374.
- [9] D. GOLDFARB AND J. K. REID, *A practical steepest-edge simplex algorithm*, Math. Programming, 12 (1977), pp. 361–371.
- [10] H. GREENBERG AND J. E. KALAN, *An exact update for Harris' tread*, Programming Studies, 4 (1975), pp. 26–29.
- [11] U. H. SUHL, *MOPS – Mathematical Optimization System*, European J. Operational Research, 72 (1994), pp. 312–322.
- [12] J. H. WILKINSON, *The Algebraic Eigenvalue Problem*, Oxford University Press, Oxford, 1965.
- [13] P. WOLFE, *A technique for resolving degeneracy in linear programming*, J. Soc. Indust. Appl. Math., 11 (1963), pp. 205–211.

BFGS WITH UPDATE SKIPPING AND VARYING MEMORY*

TAMARA G. KOLDA[†], DIANNE P. O'LEARY[‡], AND LARRY NAZARETH[§]

Abstract. We give conditions under which limited-memory quasi-Newton methods with exact line searches will terminate in n steps when minimizing n -dimensional quadratic functions. We show that although all Broyden family methods terminate in n steps in their full-memory versions, only BFGS does so with limited-memory. Additionally, we show that full-memory Broyden family methods with exact line searches terminate in at most $n + p$ steps when p matrix updates are skipped. We introduce new limited-memory BFGS variants and test them on nonquadratic minimization problems.

Key words. minimization, quasi-Newton, BFGS, limited-memory, update skipping, Broyden family

AMS subject classifications. 65K10, 65H10

PII. S1052623496306450

1. Introduction. The quasi-Newton family of algorithms remains a standard workhorse for minimization. Many of these methods share the properties of finite termination on strictly convex quadratic functions, a linear or superlinear rate of convergence on general convex functions, and no need to store or evaluate the second derivative matrix. In general, an approximation to the second derivative matrix is built by accumulating the results of earlier steps. Descriptions of many quasi-Newton algorithms can be found in books by Luenberger [17] and Dennis and Schnabel [8].

Although there are an infinite number of quasi-Newton methods, one method surpasses the others in popularity: the BFGS algorithm of Broyden, Fletcher, Goldfarb, and Shanno; see, e.g., Dennis and Schnabel [8]. This method exhibits more robust behavior than its relatives. Many attempts have been made to explain this robustness, but a complete understanding has yet to be obtained [24]. One result of the work in this paper is a small step toward this understanding, since we investigate the question of how much and which information can be dropped in BFGS and other quasi-Newton methods without destroying the property of quadratic termination.

We answer this question in the context of exact line search methods, those that find a minimizer on a one-dimensional subspace at every iteration. (In practice, inexact line searches that satisfy side conditions such as those proposed by Wolfe (see section 4.3) are substituted for exact line searches.) We focus on modifications of well-known quasi-Newton algorithms resulting from limiting the memory, either by discarding the results of early steps (section 2) or by skipping some updates to the second derivative approximation (section 3). We give conditions under which quasi-Newton methods will terminate in n steps when minimizing quadratic functions of n variables. Although all Broyden family methods (see section 2) terminate in n steps

*Received by the editors July 10, 1996; accepted for publication (in revised form) October 3, 1997; published electronically September 23, 1998.

<http://www.siam.org/journals/siopt/8-4/30645.html>

[†]Computational Methods, Building 6012, P.O. Box 2008, Oak Ridge National Laboratory, Oak Ridge, TN 37831-6367 (kolda@msr.epm.ornl.gov). This work was supported in part by the National Physical Science Consortium, the National Security Agency, and the University of Maryland.

[‡]Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742 (oleary@cs.umd.edu). This work was supported by the National Science Foundation under grant NSF CCR-95-03126.

[§]Department of Pure and Applied Mathematics, Washington State University, Pullman, WA 99164 (nazareth@amath.washington.edu).

in their full-memory versions, we show that only BFGS has n -step termination under limited-memory. We also show that the methods from the Broyden family terminate in $n + p$ steps even if p updates are skipped, but termination is lost if we both skip updates and limit the memory.

In section 4, we report the results of experiments with new limited-memory BFGS(L-BFGS) variants on problems taken from the constrained and unconstrained testing environment (CUTE) [3] test set, showing that some savings in time can be achieved.

Notation. Matrices and vectors are denoted by boldface uppercase and lowercase letters, respectively. Scalars are denoted by Greek or Roman letters. The superscript “T” denotes transposition. Subscripts denote iteration number. Products are always taken from left to right:

$$\prod_{i=j}^k \mathbf{B}_i = \begin{cases} \mathbf{B}_j \cdot \mathbf{B}_{j+1} \cdots \mathbf{B}_k & \text{if } j \leq k, \\ \mathbf{I} & \text{otherwise.} \end{cases}$$

The notation $\text{span}\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$ denotes the subspace spanned by $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k$. Whenever we refer to an n -dimensional strictly convex quadratic function, we assume it is of the form

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} - \mathbf{x}^T \mathbf{b},$$

where $\mathbf{A}$ is a positive definite $n \times n$ matrix and $\mathbf{b}$ is an n -vector.

2. Limited-memory variations of quasi-Newton algorithms. In this section we characterize full-memory and limited-memory methods that terminate in n iterations on n -dimensional strictly convex quadratic minimization problems using exact line searches. Most full-memory versions of the methods we will discuss are known to terminate in n iterations. Limited-memory methods store the quasi-Newton matrix implicitly and require less memory; furthermore, the computation of the search direction is often less expensive since it involves the implicitly stored matrix. L-BFGS was shown by Nocedal [23] to terminate in n steps. The preconditioned conjugate gradient method, which can be cast as a limited-memory quasi-Newton method, is also known to terminate in n iterations; see, e.g., Luenberger [17] or Golub and Van Loan [12]. Little else is known about termination of limited-memory methods.

Let $f(\mathbf{x})$ denote the strictly convex quadratic function to be minimized, and let $\mathbf{g}(\mathbf{x})$ denote the gradient of f . We define $\mathbf{g}_k \equiv \mathbf{g}(\mathbf{x}_k)$, where $\mathbf{x}_k$ is the k th iterate and denote the change in iterate and gradient by

$$\begin{aligned} \mathbf{s}_k &= \mathbf{x}_{k+1} - \mathbf{x}_k, \\ \mathbf{y}_k &= \mathbf{g}_{k+1} - \mathbf{g}_k. \end{aligned}$$

We present a general result that characterizes quasi-Newton methods (see Figure 2.1) that terminate in n iterations. We restrict ourselves to methods with an update of the form

$$(2.1) \quad \mathbf{H}_{k+1} = \gamma_k \mathbf{P}_k^T \mathbf{H}_0 \mathbf{Q}_k + \sum_{i=1}^{m_k} \mathbf{w}_{ik} \mathbf{z}_{ik}^T.$$

Here,

1. $\mathbf{H}_0$ is an $n \times n$ symmetric positive definite matrix that remains constant for all k , and γ_k is a nonzero scalar that can be thought of as an iterative rescaling of $\mathbf{H}_0$;

Let $\mathbf{x}_0$ be the starting point, and let $\mathbf{H}_0$ be the initial inverse Hessian approximation.
 For $k = 0, 1, \dots$

1. Compute $\mathbf{d}_k = -\mathbf{H}_k \mathbf{g}_k$.
2. Choose $\alpha_k > 0$ such that $f(\mathbf{x}_k + \alpha \mathbf{d}_k) \geq f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$ for all $\alpha > 0$.
3. Set $\mathbf{s}_k = \alpha_k \mathbf{d}_k$.
4. Set $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$.
5. Compute $\mathbf{g}_{k+1}$.
6. Set $\mathbf{y}_k = \mathbf{g}_{k+1} - \mathbf{g}_k$.
7. Choose $\mathbf{H}_{k+1}$.

FIG. 2.1. *General quasi-Newton method.*

2. $\mathbf{P}_k$ is an $n \times n$ matrix that is the product of projection matrices of the form

$$(2.2) \quad \mathbf{I} - \frac{\mathbf{u}\mathbf{v}^T}{\mathbf{u}^T\mathbf{v}},$$

where $\mathbf{u} \in \text{span}\{\mathbf{y}_0, \dots, \mathbf{y}_k\}$ and $\mathbf{v} \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{k+1}\}^1$, and $\mathbf{Q}_k$ is an $n \times n$ matrix that is the product of projection matrices of the same form where $\mathbf{u}$ is any n -vector and $\mathbf{v} \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_k\}$;

3. m_k is a nonnegative integer, $\mathbf{w}_{ik}$ ($i = 1, 2, \dots, m_k$) is any n -vector, and $\mathbf{z}_{ik}$ ($i = 1, 2, \dots, m_k$) is any vector in $\text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_k\}$.

We refer to this form as the *general form*. The general form fits many known quasi-Newton methods, including the Broyden family and the L-BFGS method. We do not assume that these quasi-Newton methods satisfy the secant condition

$$\mathbf{H}_{k+1}\mathbf{y}_k = \mathbf{s}_k,$$

nor that $\mathbf{H}_{k+1}$ is positive definite and symmetric. Symmetric positive definite updates are desirable since this guarantees that the quasi-Newton method produces descent directions. Note that if the update is not positive definite, we may produce a $\mathbf{d}_k$ such that $\mathbf{d}_k^T \mathbf{g}_k > 0$, in which case we choose α_k over all *negative* α rather than all positive α .

Example 1. The method of steepest descent [17] fits the general form (2.1). For each k we define

$$(2.3) \quad \gamma_k = 1, \quad m_k = 0, \quad \text{and} \quad \mathbf{P}_k = \mathbf{Q}_k = \mathbf{H}_0 = \mathbf{I}.$$

Note that neither $\mathbf{w}$ nor $\mathbf{z}$ vectors are specified since $m_k = 0$.

Example 2. The $(k+1)$ st update for the conjugate gradient method with preconditioner $\mathbf{H}_0$ fits the general form (2.1) with

$$(2.4) \quad \gamma_k = 1, \quad m_k = 0, \quad \mathbf{P}_k = \mathbf{I} - \frac{\mathbf{y}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{y}_k}, \quad \text{and} \quad \mathbf{Q}_k = \mathbf{I}.$$

Example 3. A full-memory quasi-Newton method can be converted into a limited-memory method in the following way. Define $\mathbf{H}_{k+1}$ to be the result of applying the

¹The vector $\mathbf{s}_{k+1}$ has not yet been explicitly calculated but is needed here only for the theoretical framework, not for the computational algorithms. In fact, it may also be available computationally in algorithms such as the limited memory DFP; see the proof of Proposition 2.1.

update formula to $\mathbf{H}_0$ m times using the m most recent $(\mathbf{s}, \mathbf{y})$ pairs. The L-BFGS update (see Nocedal [23]) with limited-memory constant m can be written as

$$(2.5) \quad \mathbf{H}_{k+1} = \mathbf{V}_{k-m_k+1,k}^T \mathbf{H}_0 \mathbf{V}_{k-m_k+1,k} + \sum_{i=k-m_k+1}^k \mathbf{V}_{i+1,k}^T \frac{\mathbf{s}_i \mathbf{s}_i^T}{\mathbf{s}_i^T \mathbf{y}_i} \mathbf{V}_{i+1,k},$$

where $m_k = \min\{k+1, m\}$ and

$$\mathbf{V}_{ik} = \prod_{j=i}^k \left(\mathbf{I} - \frac{\mathbf{y}_j \mathbf{s}_j^T}{\mathbf{s}_j^T \mathbf{y}_j} \right).$$

L-BFGS fits the general form (2.1) if at iteration k we choose

$$(2.6) \quad \begin{aligned} \gamma_k &= 1, \quad m_k = \min\{k+1, m\}, \\ \mathbf{P}_k &= \mathbf{Q}_k = \mathbf{V}_{k-m_k+1,k}, \quad \text{and} \\ \mathbf{w}_{ik} &= \mathbf{z}_{ik} = \frac{(\mathbf{V}_{k-m_k+i+1,k})^T (\mathbf{s}_{k-m_k+i})}{\sqrt{(\mathbf{s}_{k-m_k+i})^T (\mathbf{y}_{k-m_k+i})}}. \end{aligned}$$

Observe that $\mathbf{P}_k$, $\mathbf{Q}_k$, and $\mathbf{z}_{ik}$ all obey the constraints imposed on their construction.

Example 4. We define limited-memory DFP (L-DFP) in a similar way: $\mathbf{H}_{k+1} = \hat{\mathbf{H}}_{k+1}^{(m_k)}$, where, for $i = 0, \dots, m_k$,

$$\hat{\mathbf{H}}_{k+1}^{(0)} = \mathbf{H}_0$$

and

$$\hat{\mathbf{H}}_{k+1}^{(i)} = \hat{\mathbf{H}}_{k+1}^{(i-1)} + \mathbf{U}_{\text{DFP}}(\hat{\mathbf{H}}_{k+1}^{(i-1)}, \mathbf{s}_{k-m_k+i}, \mathbf{y}_{k-m_k+i}),$$

with

$$\mathbf{U}_{\text{DFP}}(\mathbf{H}, \mathbf{s}, \mathbf{y}) = -\frac{\mathbf{H} \mathbf{y} \mathbf{y}^T \mathbf{H}}{\mathbf{y}^T \mathbf{H} \mathbf{y}} + \frac{\mathbf{s} \mathbf{s}^T}{\mathbf{s}^T \mathbf{y}}.$$

To simplify our description, note that $\hat{\mathbf{H}}_{k+1}^{(i)}$ can be rewritten as

$$\begin{aligned} \hat{\mathbf{H}}_{k+1}^{(i)} &= \left(\mathbf{I} - \frac{\hat{\mathbf{H}}_{k+1}^{(i-1)} \mathbf{y}_{k-m_k+i} \mathbf{y}_{k-m_k+i}^T}{\mathbf{y}_{k-m_k+i}^T \hat{\mathbf{H}}_{k+1}^{(i-1)} \mathbf{y}_{k-m_k+i}} \right) \hat{\mathbf{H}}_{k+1}^{(i-1)} + \frac{\mathbf{s}_{k-m_k+i} \mathbf{s}_{k-m_k+i}^T}{\mathbf{s}_{k-m_k+i}^T \mathbf{y}_{k-m_k+i}} \\ &= (\hat{\mathbf{V}}_{0k}^{(i)})^T \mathbf{H}_0 + \sum_{j=1}^i (\hat{\mathbf{V}}_{jk}^{(i)})^T \frac{\mathbf{s}_{k-m_k+j} \mathbf{s}_{k-m_k+j}^T}{\mathbf{s}_{k-m_k+j}^T \mathbf{y}_{k-m_k+j}}, \end{aligned}$$

for $i \geq 1$, where

$$\hat{\mathbf{V}}_{jk}^{(i)} = \prod_{l=j+1}^i \left[\mathbf{I} - \frac{\mathbf{y}_{k-m_k+l} (\mathbf{H}_{k+1}^{(l-1)} \mathbf{y}_{k-m_k+l})^T}{\mathbf{y}_{k-m_k+l}^T \mathbf{H}_{k+1}^{(l-1)} \mathbf{y}_{k-m_k+l}} \right].$$

Thus $\mathbf{H}_{k+1}$ can be written as

$$(2.7) \quad \mathbf{H}_{k+1} = \mathbf{V}_{0k}^T \mathbf{H}_0 + \sum_{i=1}^{m_k} \left(\mathbf{V}_{ik}^T \frac{\mathbf{s}_{k-m_k+i} \mathbf{s}_{k-m_k+i}^T}{\mathbf{s}_{k-m_k+i}^T \mathbf{y}_{k-m_k+i}} \right),$$

where

$$\mathbf{V}_{ik} = \prod_{j=i+1}^{m_k} \left[\mathbf{I} - \frac{\mathbf{y}_{k-m_k+j} \left(\hat{\mathbf{H}}_{k+1}^{(j-1)} \mathbf{y}_{k-m_k+j} \right)^T}{\mathbf{y}_{k-m_k+j}^T \hat{\mathbf{H}}_{k+1}^{(j-1)} \mathbf{y}_{k-m_k+j}} \right].$$

Equation (2.7) looks very much like the general form given in (2.1). L-DFP fits the general form with the following choices:

$$(2.8) \quad \begin{aligned} \gamma_k &= 1, \quad \mathbf{P}_k = \mathbf{V}_{0k}, \quad \mathbf{Q}_k = \mathbf{I}, \\ \mathbf{w}_{ik} &= \mathbf{V}_{ik}^T \mathbf{s}_{k-m_k+i} / (\mathbf{s}_{k-m_k+i}^T \mathbf{y}_{k-m_k+i}), \text{ and } \mathbf{z}_{ik} = \mathbf{s}_{k-m_k+i}. \end{aligned}$$

Except for the choice of $\mathbf{P}_k$, it is trivial to verify that the choices satisfy the general form. To prove that $\mathbf{P}_k$ satisfies the requirements, we need to show

$$(2.9) \quad \hat{\mathbf{H}}_{k+1}^{(i-1)} \mathbf{y}_{k-m_k+i} \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{k+1}\}, \text{ for } i = 1, \dots, m_k \text{ and all } k.$$

PROPOSITION 2.1. *For L-DFP, the following three conditions hold for each value of k :*

$$(2.10) \quad \hat{\mathbf{H}}_{k+1}^{(i-1)} \mathbf{y}_{k-m_k+i} \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_k\} \text{ for } i = 1, \dots, m_k - 1,$$

$$(2.11) \quad \hat{\mathbf{H}}_{k+1}^{(i-1)} \mathbf{y}_{k-m_k+i} \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_k, \mathbf{H}_0 \mathbf{g}_{k+1}\} \text{ for } i = m_k, \text{ and}$$

$$(2.12) \quad \text{span}\{\mathbf{H}_0 \mathbf{g}_0, \dots, \mathbf{H}_0 \mathbf{g}_{k+1}\} \subseteq \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{k+1}\}.$$

Proof. We will prove this via induction. Suppose $k = 0$. Then $m_0 = 1$. We have

$$\hat{\mathbf{H}}_{k+1}^{(0)} \mathbf{y}_k = \mathbf{H}_0 \mathbf{y}_0 = \mathbf{H}_0 \mathbf{g}_1 - \mathbf{H}_0 \mathbf{g}_0 \in \text{span}\{\mathbf{s}_0, \mathbf{H}_0 \mathbf{g}_1\}.$$

(Recall that $\text{span}\{\mathbf{s}_0\}$ is trivially equal to $\text{span}\{\mathbf{H}_0 \mathbf{g}_0\}$.) Furthermore,

$$\begin{aligned} \mathbf{s}_1 &= -\alpha_1 \mathbf{H}_1 \mathbf{g}_1 \\ &= -\alpha_1 \left[\mathbf{H}_0 \mathbf{g}_1 - \frac{\mathbf{y}_0^T \mathbf{H}_0 \mathbf{g}_1}{\mathbf{y}_0^T \mathbf{H}_0 \mathbf{y}_0} (\mathbf{H}_0 \mathbf{g}_1 - \mathbf{H}_0 \mathbf{g}_0) + \frac{\mathbf{s}_0^T \mathbf{g}_1}{\mathbf{y}_0^T \mathbf{s}_0} \mathbf{s}_0 \right]. \end{aligned}$$

So we can conclude

$$\left(1 - \frac{\mathbf{y}_0^T \mathbf{H}_0 \mathbf{g}_1}{\mathbf{y}_0^T \mathbf{H}_0 \mathbf{y}_0} \right) \mathbf{H}_0 \mathbf{g}_1 = - \left[\frac{1}{\alpha_1} \mathbf{s}_1 + \frac{\mathbf{y}_0^T \mathbf{H}_0 \mathbf{g}_1}{\mathbf{y}_0^T \mathbf{H}_0 \mathbf{y}_0} \mathbf{H}_0 \mathbf{g}_0 + \frac{\mathbf{s}_0^T \mathbf{g}_1}{\mathbf{y}_0^T \mathbf{s}_0} \mathbf{s}_0 \right].$$

Hence, $\mathbf{H}_0 \mathbf{g}_1 \in \text{span}\{\mathbf{s}_0, \mathbf{s}_1\}$, and so the base case holds.

Assume that

$$\begin{aligned} \hat{\mathbf{H}}_k^{(i-1)} \mathbf{y}_{k-1-m_{k-1}+i} &\in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{k-1}\} \text{ for } i = 1, \dots, m_{k-1} - 1, \\ \hat{\mathbf{H}}_k^{(i-1)} \mathbf{y}_{k-1-m_{k-1}+i} &\in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{k-1}, \mathbf{H}_0 \mathbf{g}_k\} \text{ for } i = m_{k-1}, \text{ and} \\ \text{span}\{\mathbf{H}_0 \mathbf{g}_0, \dots, \mathbf{H}_0 \mathbf{g}_k\} &\subseteq \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_k\}. \end{aligned}$$

Using the induction assumption, we will show that (2.10)–(2.12) holds for $(k+1)$. We show (2.10) for $i = 1, \dots, m_k - 1$. For $i = 1$ (assume $m_k > i$),

$$\hat{\mathbf{H}}_{k+1}^{(0)} \mathbf{y}_{k-m_k+1} = \mathbf{H}_0 \mathbf{y}_{k-m_k+1} = \mathbf{H}_0 \mathbf{g}_{k-m_k+2} - \mathbf{H}_0 \mathbf{g}_{k-m_k+1}.$$

Using the induction hypothesis, we get that

$$\hat{\mathbf{H}}_{k+1}^{(0)} \mathbf{y}_{k-m_k+1} \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_k\}.$$

Assume that

$$(2.13) \quad \hat{\mathbf{H}}_{k+1}^{(j)} \mathbf{y}_{k-m_k+j+1} \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_k\}$$

for j between 1 and $i-2$, $i \leq m_k - 2$. Then

$$\begin{aligned} \hat{\mathbf{H}}_{k+1}^{(i-1)} \mathbf{y}_{k-m_k+i} &= \left(\hat{\mathbf{V}}_{0k}^{(i-1)} \right)^T \mathbf{H}_0 \mathbf{y}_{k-m_k+i} \\ &\quad + \sum_{j=1}^{i-1} \frac{\mathbf{s}_{k-m_k+j-1}^T \mathbf{y}_{k-m_k+i}}{\mathbf{s}_{k-m_k+j-1}^T \mathbf{y}_{k-m_k+j-1}} \left(\hat{\mathbf{V}}_{jk}^{(i-1)} \right)^T \mathbf{s}_{k-m_k+j-1}. \end{aligned}$$

For values of $i \leq m_k - 1$, $\left(\hat{\mathbf{V}}_{jk}^{(i-1)} \right)^T$ maps any vector $\mathbf{v}$ into

$$\text{span}\{\mathbf{v}, \hat{\mathbf{H}}_{k+1}^{(0)} \mathbf{y}_{k-m_k+1}, \dots, \hat{\mathbf{H}}_{k+1}^{(i-2)} \mathbf{y}_{k-2}\},$$

and so $\hat{\mathbf{H}}_{k+1}^{(i-1)} \mathbf{y}_{k-m_k+i}$ is in

$$\text{span}\{\mathbf{H}_0 \mathbf{y}_{k-m_k+i}, \hat{\mathbf{H}}_{k+1}^{(0)} \mathbf{y}_{k-m_k+1}, \dots, \hat{\mathbf{H}}_{k+1}^{(i-2)} \mathbf{y}_{k-2}, \mathbf{s}_{k-m_k+1}, \dots, \mathbf{s}_{k-2}\}.$$

Using the induction hypothesis and (2.13), we get

$$\hat{\mathbf{H}}_{k+1}^{(i-1)} \mathbf{y}_{k-m_k+i} \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_k\},$$

and we can conclude that (2.10) is true for $i = 1, \dots, m_k - 1$ in the $(k+1)$ st case. If $i = m_k$, then

$$\hat{\mathbf{H}}_{k+1}^{(m_k-1)} \mathbf{y}_k \in \text{span}\{\mathbf{H}_0 \mathbf{y}_k, \hat{\mathbf{H}}_{k+1}^{(0)} \mathbf{y}_{k-m_k+1}, \dots, \hat{\mathbf{H}}_{k+1}^{(m_k-2)} \mathbf{y}_{k-1}, \mathbf{s}_{k-m_k+1}, \dots, \mathbf{s}_{k-1}\},$$

so

$$\hat{\mathbf{H}}_{k+1}^{(m_k-1)} \mathbf{y}_k \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_k, \mathbf{H}_0 \mathbf{g}_{k+1}\}.$$

Hence (2.11) is true for $(k+1)$.

Now, consider

$$\begin{aligned} \mathbf{s}_{k+1} &= -\alpha_{k+1} \mathbf{H}_{k+1} \mathbf{g}_{k+1} \\ &= \mathbf{V}_{0k}^T \mathbf{H}_0 \mathbf{g}_{k+1} + \sum_{i=1}^{m_k} \frac{\mathbf{s}_{k-m_k+i}^T \mathbf{g}_{k+1}}{\mathbf{s}_{k-m_k+i}^T \mathbf{y}_{k-m_k+i}} \mathbf{V}_{ik}^T \mathbf{s}_{k-m_k+i}. \end{aligned}$$

Using the structure of $\mathbf{V}_{jk}$ and (2.10) we see that $\mathbf{H}_0 \mathbf{g}_{k+1} \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{k+1}\}$. Hence, (2.12) also holds for $(k+1)$. $\square$

Example 5. The Broyden family is the class of quasi-Newton methods whose matrices are linear combinations of the DFP and BFGS matrices:

$$\mathbf{H}_{k+1} = \phi \mathbf{H}_{k+1}^{BFGS} + (1 - \phi) \mathbf{H}_{k+1}^{DFP}, \quad \phi \in \mathbf{R};$$

see, e.g., Luenberger [17, Chap. 9]. The parameter ϕ is usually restricted to values that are guaranteed to produce a positive definite update, although recent work with SR1, a Broyden family method, by Khalfan, Byrd, and Schnabel [15] may change this practice. No restriction on ϕ is necessary for the development of our theory. The Broyden family update can be expressed as

$$\begin{aligned} \mathbf{H}_{k+1} = & \mathbf{H}_k + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{y}_k} - \frac{\mathbf{H}_k \mathbf{y}_k \mathbf{y}_k^T \mathbf{H}_k}{\mathbf{y}_k^T \mathbf{H}_k \mathbf{y}_k} \\ & + \phi (\mathbf{y}_k^T \mathbf{H}_k \mathbf{y}_k) \left(\frac{\mathbf{s}_k}{\mathbf{s}_k^T \mathbf{y}_k} - \frac{\mathbf{H}_k \mathbf{y}_k}{\mathbf{y}_k^T \mathbf{H}_k \mathbf{y}_k} \right) \left(\frac{\mathbf{s}_k}{\mathbf{s}_k^T \mathbf{y}_k} - \frac{\mathbf{H}_k \mathbf{y}_k}{\mathbf{y}_k^T \mathbf{H}_k \mathbf{y}_k} \right)^T. \end{aligned}$$

We sketch the explanation of how the full-memory version fits the general form given in (2.1). The limited-memory case is similar. We can rewrite the Broyden family update as follows:

$$\begin{aligned} \mathbf{H}_{k+1} = & \mathbf{H}_k + (\phi - 1) \frac{\mathbf{H}_k \mathbf{y}_k \mathbf{y}_k^T \mathbf{H}_k}{\mathbf{y}_k^T \mathbf{H}_k \mathbf{y}_k} - \phi \frac{\mathbf{s}_k \mathbf{y}_k^T}{\mathbf{s}_k^T \mathbf{y}_k} \mathbf{H}_k + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{y}_k} \\ & + \phi \frac{\mathbf{y}_k^T \mathbf{H}_k \mathbf{y}_k \cdot \mathbf{s}_k \mathbf{s}_k^T}{(\mathbf{s}_k^T \mathbf{y}_k)^2} - \phi \frac{\mathbf{H}_k \mathbf{y}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{y}_k} \\ = & \left[\mathbf{I} - \frac{((1 - \phi) \mathbf{s}_k^T \mathbf{y}_k \cdot \mathbf{H}_k \mathbf{y}_k + \phi \mathbf{y}_k^T \mathbf{H}_k \mathbf{y}_k \cdot \mathbf{s}_k) \mathbf{y}_k^T}{\mathbf{y}_k^T \mathbf{H}_k \mathbf{y}_k \cdot \mathbf{s}_k^T \mathbf{y}_k} \right] \mathbf{H}_k \\ & + \left[\left(1 + \phi \frac{\mathbf{y}_k^T \mathbf{H}_k \mathbf{y}_k}{\mathbf{s}_k^T \mathbf{y}_k} \right) \mathbf{s}_k - \phi \mathbf{H}_k \mathbf{y}_k \right] \frac{\mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{y}_k}. \end{aligned}$$

Hence,

$$\mathbf{H}_{k+1} = \mathbf{V}_{0k} \mathbf{H}_0 + \sum_{i=1}^{k+1} \mathbf{w}_{ik} \mathbf{z}_{ik}^T,$$

where

$$\begin{aligned} \mathbf{V}_{ik} = & \prod_{j=i}^k \left[\mathbf{I} - \frac{((1 - \phi) \mathbf{s}_j^T \mathbf{y}_j \cdot \mathbf{H}_j \mathbf{y}_j + \phi \mathbf{y}_j^T \mathbf{H}_j \mathbf{y}_j \cdot \mathbf{s}_j) \mathbf{y}_j^T}{\mathbf{y}_j^T \mathbf{H}_j \mathbf{y}_j \cdot \mathbf{s}_j^T \mathbf{y}_j} \right], \\ \mathbf{w}_{ik} = & \mathbf{V}_{ik} \left[\left(1 + \phi \frac{\mathbf{y}_{i-1}^T \mathbf{H}_{i-1} \mathbf{y}_{i-1}}{\mathbf{s}_{i-1}^T \mathbf{y}_{i-1}} \mathbf{s}_{i-1} \right) - \phi \mathbf{H}_{i-1} \mathbf{y}_{i-1} \right], \text{ and } \mathbf{z}_{ik} = \frac{\mathbf{s}_{i-1}^T}{\mathbf{s}_{i-1}^T \mathbf{y}_{i-1}}. \end{aligned}$$

It is left to the reader to show that $\mathbf{H}_k \mathbf{y}_k$ is in $\text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{k+1}\}$, and thus the Broyden family updates fit the form in (2.1).

2.1. Termination of limited-memory methods. In this section we show that methods fitting the general form (2.1) produce conjugate search directions (see Theorem 2.2) and terminate in n iterations (see Corollary 2.3) *if and only if* $\mathbf{P}_k$ maps the vectors $\mathbf{y}_0$ through $\mathbf{y}_k$ into $\text{span}\{\mathbf{y}_0, \dots, \mathbf{y}_{k-1}\}$ for each $k = 1, 2, \dots, n$. Furthermore, this condition on $\mathbf{P}_k$ is satisfied only if $\mathbf{y}_k$ is used in its formation (see Corollary 2.4).

THEOREM 2.2. *Suppose that we apply a quasi-Newton method (Figure 2.1) with an update of the form (2.1) to minimize an n -dimensional strictly convex quadratic function. Then for each k before termination (i.e., $\mathbf{g}_{k+1} \neq 0$),*

$$(2.14) \quad \mathbf{g}_{k+1}^T \mathbf{s}_j = 0, \text{ for all } j = 0, 1, \dots, k,$$

$$(2.15) \quad \mathbf{s}_{k+1}^T \mathbf{A} \mathbf{s}_j = 0, \text{ for all } j = 0, 1, \dots, k, \text{ and}$$

$$(2.16) \quad \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{k+1}\} = \text{span}\{\mathbf{H}_0 \mathbf{g}_0, \dots, \mathbf{H}_0 \mathbf{g}_{k+1}\},$$

if and only if

$$(2.17) \quad \mathbf{P}_j \mathbf{y}_i \in \text{span}\{\mathbf{y}_0, \dots, \mathbf{y}_{j-1}\} \text{ for all } i = 0, 1, \dots, j, \quad j = 0, 1, \dots, k.$$

Proof. ($\Leftarrow$) Assume that (2.17) holds. We will prove (2.14)–(2.16) by induction. Since the line searches are exact, $\mathbf{g}_1$ is orthogonal to $\mathbf{s}_0$. Using the fact that $\mathbf{P}_0 \mathbf{y}_0 = 0$ from (2.17) and the fact that $\mathbf{z}_{i0} \in \text{span}\{\mathbf{s}_0\}$ implies $\mathbf{g}_1^T \mathbf{z}_{i0} = 0$, $i = 1, \dots, m_k$, we see that $\mathbf{s}_1$ is conjugate to $\mathbf{s}_0$ since

$$\begin{aligned} \mathbf{s}_1^T \mathbf{A} \mathbf{s}_0 &= \alpha_1 \mathbf{d}_1^T \mathbf{y}_0 \\ &= -\alpha_1 \mathbf{g}_1^T \mathbf{H}_1^T \mathbf{y}_0 \\ &= -\alpha_1 \mathbf{g}_1^T \left(\gamma_0 \mathbf{Q}_0^T \mathbf{H}_0 \mathbf{P}_0 + \sum_{i=0}^{m_0} \mathbf{z}_{i0} \mathbf{w}_{i0}^T \right) \mathbf{y}_0 \\ &= 0. \end{aligned}$$

Finally, $\text{span}\{\mathbf{s}_0\} = \text{span}\{\mathbf{H}_0 \mathbf{g}_0\}$, and so the base case is established.

We will assume that claims (2.14)–(2.16) hold for $k = 0, 1, \dots, \hat{k} - 1$ and prove that they also hold for $k = \hat{k}$.

The vector $\mathbf{g}_{\hat{k}+1}$ is orthogonal to $\mathbf{s}_{\hat{k}}$ since the line search is exact. Using the induction hypotheses that $\mathbf{g}_{\hat{k}}$ is orthogonal to $\{\mathbf{s}_0, \dots, \mathbf{s}_{\hat{k}-1}\}$ and $\mathbf{s}_{\hat{k}}$ is conjugate to $\{\mathbf{s}_0, \dots, \mathbf{s}_{\hat{k}-1}\}$, we see that, for $j < \hat{k}$,

$$\mathbf{g}_{\hat{k}+1}^T \mathbf{s}_j = (\mathbf{g}_{\hat{k}} + \mathbf{y}_{\hat{k}})^T \mathbf{s}_j = (\mathbf{g}_{\hat{k}} + \mathbf{A} \mathbf{s}_{\hat{k}})^T \mathbf{s}_j = 0.$$

Hence, (2.14) holds for $k = \hat{k}$.

To prove (2.15), we note that

$$\mathbf{s}_{\hat{k}+1}^T \mathbf{A} \mathbf{s}_j = -\alpha_{\hat{k}+1} \mathbf{g}_{\hat{k}+1}^T \mathbf{H}_{\hat{k}+1}^T \mathbf{y}_j,$$

so it is sufficient to prove that $\mathbf{g}_{\hat{k}+1}^T \mathbf{H}_{\hat{k}+1}^T \mathbf{y}_j = 0$ for $j = 0, 1, \dots, \hat{k}$. We will use the following facts:

(i) $\mathbf{g}_{\hat{k}+1}^T \mathbf{Q}_{\hat{k}}^T = \mathbf{g}_{\hat{k}+1}^T$ since the $\mathbf{v}$ in each of the projections used to form $\mathbf{Q}_{\hat{k}}$ is in $\text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{\hat{k}}\}$, and $\mathbf{g}_{\hat{k}+1}$ is orthogonal to that span.

(ii) $\mathbf{g}_{\hat{k}+1}^T \mathbf{z}_{i\hat{k}} = 0$ for $i = 1, \dots, m_{\hat{k}}$ since each $\mathbf{z}_{i\hat{k}}$ is in $\text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{\hat{k}}\}$, and $\mathbf{g}_{\hat{k}+1}$ is orthogonal to that span.

(iii) Since we are assuming that (2.17) holds true, for each $j = 0, 1, \dots, \hat{k}$ there exist $\mu_0, \dots, \mu_{\hat{k}-1}$ such that $\mathbf{P}_{\hat{k}} \mathbf{y}_j$ can be expressed as $\sum_{i=0}^{\hat{k}-1} \mu_i \mathbf{y}_i$.

(iv) For $i = 0, 1, \dots, \hat{k}-1$, $\mathbf{g}_{\hat{k}+1}$ is orthogonal to $\mathbf{H}_0 \mathbf{y}_i$ because $\mathbf{g}_{\hat{k}+1}$ is orthogonal to $\text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{\hat{k}}\}$ and $\mathbf{H}_0 \mathbf{y}_i \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{\hat{k}}\}$ from (2.16).

Thus,

$$\begin{aligned} \mathbf{g}_{\hat{k}+1}^T \mathbf{H}_{\hat{k}+1}^T \mathbf{y}_j &= \mathbf{g}_{\hat{k}+1}^T \left(\gamma_{\hat{k}} \mathbf{Q}_{\hat{k}}^T \mathbf{H}_0 \mathbf{P}_{\hat{k}} + \sum_{i=1}^{m_{\hat{k}}} \mathbf{z}_{i\hat{k}} \mathbf{w}_{i\hat{k}}^T \right) \mathbf{y}_j \\ &= \gamma_{\hat{k}} \mathbf{g}_{\hat{k}+1}^T \mathbf{Q}_{\hat{k}}^T \mathbf{H}_0 \mathbf{P}_{\hat{k}} \mathbf{y}_j + \sum_{i=1}^{m_{\hat{k}}} \mathbf{g}_{\hat{k}+1}^T \mathbf{z}_{i\hat{k}} \mathbf{w}_{i\hat{k}}^T \mathbf{y}_j \\ &= \gamma_{\hat{k}} \mathbf{g}_{\hat{k}+1}^T \mathbf{H}_0 \mathbf{P}_{\hat{k}} \mathbf{y}_j \end{aligned}$$

$$\begin{aligned}
&= \gamma_{\hat{k}} \mathbf{g}_{\hat{k}+1}^T \mathbf{H}_0 \left(\sum_{i=1}^{\hat{k}-1} \mu_i \mathbf{y}_i \right) \\
&= \gamma_{\hat{k}} \sum_{i=1}^{\hat{k}-1} \mu_i \mathbf{g}_{\hat{k}+1}^T \mathbf{H}_0 \mathbf{y}_i \\
&= 0.
\end{aligned}$$

Thus, (2.15) holds for $k = \hat{k}$.

Finally, using (i) and (ii) from above,

$$\begin{aligned}
\mathbf{s}_{\hat{k}+1} &= -\alpha_{\hat{k}+1} \mathbf{H}_{\hat{k}+1} \mathbf{g}_{\hat{k}+1} \\
&= -\alpha_{\hat{k}+1} \left(\gamma_{\hat{k}} \mathbf{P}_{\hat{k}}^T \mathbf{H}_0 \mathbf{Q}_{\hat{k}} \mathbf{g}_{\hat{k}+1} + \sum_{i=1}^{m_{\hat{k}}} \mathbf{w}_{i\hat{k}} \mathbf{z}_{i\hat{k}}^T \mathbf{g}_{\hat{k}+1} \right) \\
&= -\alpha_{\hat{k}+1} \gamma_{\hat{k}} \mathbf{P}_{\hat{k}}^T \mathbf{H}_0 \mathbf{g}_{\hat{k}+1}.
\end{aligned}$$

Since $\mathbf{P}_{\hat{k}}^T$ maps any vector $\mathbf{v}$ into $\text{span}\{\mathbf{v}, \mathbf{s}_0, \dots, \mathbf{s}_{\hat{k}+1}\}$ by construction, there exist $\sigma_0, \dots, \sigma_{\hat{k}+1}$ such that

$$\mathbf{s}_{\hat{k}+1} = -\alpha_{\hat{k}+1} \gamma_{\hat{k}} \left(\mathbf{H}_0 \mathbf{g}_{\hat{k}+1} + \sum_{i=0}^{\hat{k}+1} \sigma_i \mathbf{s}_i \right).$$

Hence,

$$\mathbf{H}_0 \mathbf{g}_{\hat{k}+1} \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{\hat{k}+1}\},$$

so

$$\text{span}\{\mathbf{H}_0 \mathbf{g}_0, \dots, \mathbf{H}_0 \mathbf{g}_{\hat{k}+1}\} \subseteq \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{\hat{k}+1}\}.$$

To show equality of the sets, we will show that $\mathbf{H}_0 \mathbf{g}_{\hat{k}+1}$ is linearly independent of $\{\mathbf{H}_0 \mathbf{g}_0, \dots, \mathbf{H}_0 \mathbf{g}_{\hat{k}}\}$. (We already know that the vectors $\mathbf{H}_0 \mathbf{g}_0, \dots, \mathbf{H}_0 \mathbf{g}_{\hat{k}}$ are linearly independent since they span the same space as the linearly independent set $\{\mathbf{s}_0, \dots, \mathbf{s}_{\hat{k}}\}$.) Suppose that $\mathbf{H}_0 \mathbf{g}_{\hat{k}+1}$ is not linearly independent. Then there exist $\phi_0, \dots, \phi_{\hat{k}}$, not all zero, such that

$$\mathbf{H}_0 \mathbf{g}_{\hat{k}+1} = \sum_{i=0}^{\hat{k}} \phi_i \mathbf{H}_0 \mathbf{g}_i.$$

Recall that $\mathbf{g}_{\hat{k}+1}$ is orthogonal to $\{\mathbf{s}_0, \dots, \mathbf{s}_{\hat{k}}\}$. By our induction assumption, this implies that $\mathbf{g}_{\hat{k}+1}$ is also orthogonal to $\{\mathbf{H}_0 \mathbf{g}_0, \dots, \mathbf{H}_0 \mathbf{g}_{\hat{k}}\}$. Thus, for any j between 0 and $\hat{k}$,

$$0 = \mathbf{g}_{\hat{k}+1}^T \mathbf{H}_0 \mathbf{g}_j = \left(\sum_{i=0}^{\hat{k}} \phi_i \mathbf{H}_0 \mathbf{g}_i \right)^T \mathbf{g}_j = \sum_{i=0}^{\hat{k}} \phi_i \mathbf{g}_i^T \mathbf{H}_0 \mathbf{g}_j = \phi_j \mathbf{g}_j^T \mathbf{H}_0 \mathbf{g}_j.$$

Since $\mathbf{H}_0$ is positive definite and $\mathbf{g}_j$ is nonzero, we conclude that ϕ_j must be zero. Since this is true for every j between zero and $\hat{k}$, we have a contradiction. Thus, the set $\{\mathbf{H}_0 \mathbf{g}_0, \dots, \mathbf{H}_0 \mathbf{g}_{\hat{k}+1}\}$ is linearly independent. Hence, (2.16) holds for $k = \hat{k}$.

($\Rightarrow$) Assume that (2.14)–(2.16) hold for all k such that $\mathbf{g}_{k+1} \neq 0$ but that (2.17) does not hold; i.e., there exist j and k such that $\mathbf{g}_{k+1} \neq 0$, j is between 0 and k , and

$$(2.18) \quad \mathbf{P}_k \mathbf{y}_j \notin \text{span}\{\mathbf{y}_0, \dots, \mathbf{y}_{k-1}\}.$$

This will lead to a contradiction. By construction of $\mathbf{P}_k$, there exist $\mu_0, \dots, \mu_k$ such that

$$(2.19) \quad \mathbf{P}_k \mathbf{y}_j = \sum_{i=0}^k \mu_i \mathbf{y}_i.$$

By assumption (2.18), μ_k must be nonzero. From (2.15), it follows that $\mathbf{g}_{k+1}^T \mathbf{H}_{k+1}^T \mathbf{y}_j = 0$. Using facts (i), (ii), and (iv) from before, (2.16), and (2.19), we get

$$\begin{aligned} 0 &= \mathbf{g}_{k+1}^T \mathbf{H}_{k+1}^T \mathbf{y}_j = \mathbf{g}_{k+1}^T \left(\gamma_k \mathbf{Q}_k^T \mathbf{H}_0 \mathbf{P}_k + \sum_{i=1}^{m_k} \mathbf{z}_{ik} \mathbf{w}_{ik}^T \right) \mathbf{y}_j \\ &= \gamma_k \mathbf{g}_{k+1}^T \mathbf{Q}_k^T \mathbf{H}_0 \mathbf{P}_k \mathbf{y}_j + \sum_{i=1}^{m_k} \mathbf{g}_{k+1}^T \mathbf{z}_{ik} \mathbf{w}_{ik}^T \mathbf{y}_j \\ &= \gamma_k \mathbf{g}_{k+1}^T \mathbf{H}_0 \mathbf{P}_k \mathbf{y}_j \\ &= \gamma_k \mathbf{g}_{k+1}^T \mathbf{H}_0 \left(\sum_{i=0}^k \mu_i \mathbf{y}_i \right) \\ &= \gamma_k \mu_k \mathbf{g}_{k+1}^T \mathbf{H}_0 \mathbf{y}_k \\ &= \gamma_k \mu_k (\mathbf{g}_{k+1}^T \mathbf{H}_0 \mathbf{g}_{k+1} - \mathbf{g}_{k+1}^T \mathbf{H}_0 \mathbf{g}_k) \\ &= \gamma_k \mu_k \mathbf{g}_{k+1}^T \mathbf{H}_0 \mathbf{g}_{k+1}. \end{aligned}$$

Thus, since neither γ_k nor μ_k is zero, we must have

$$\mathbf{g}_{k+1}^T \mathbf{H}_0 \mathbf{g}_{k+1} = 0,$$

but this is a contradiction since $\mathbf{H}_0$ is positive definite and $\mathbf{g}_{k+1}$ was assumed to be nonzero. $\square$

When a method produces conjugate search directions, we can say something about termination.

COROLLARY 2.3. *Suppose we have a method of the type described in Theorem 2.2 satisfying (2.17). Suppose further that $\mathbf{H}_k \mathbf{g}_k \neq 0$ whenever $\mathbf{g}_k \neq 0$. Then the scheme reproduces the iterates from the conjugate gradient method with preconditioner $\mathbf{H}_0$ and terminates in no more than n iterations.*

Proof. Let k be such that $\mathbf{g}_0, \dots, \mathbf{g}_k$ are all nonzero and such that $\mathbf{H}_i \mathbf{g}_i \neq 0$ for $i = 0, \dots, k$. Since we have a method of the type described in Theorem 2.2 satisfying (2.17), conditions (2.14)–(2.16) hold. We claim that the $(k+1)$ st subspace of search directions, $\text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_k\}$, is equal to the $(k+1)$ st Krylov subspace, $\text{span}\{\mathbf{H}_0 \mathbf{g}_0, \dots, (\mathbf{H}_0 \mathbf{A})^k \mathbf{H}_0 \mathbf{g}_0\}$.

From (2.16), we know that $\text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_k\} = \text{span}\{\mathbf{H}_0 \mathbf{g}_0, \dots, \mathbf{H}_0 \mathbf{g}_k\}$. We will show via induction that $\text{span}\{\mathbf{H}_0 \mathbf{g}_0, \dots, \mathbf{H}_0 \mathbf{g}_k\} = \text{span}\{\mathbf{H}_0 \mathbf{g}_0, \dots, (\mathbf{H}_0 \mathbf{A})^k \mathbf{H}_0 \mathbf{g}_0\}$. This base case is trivial, so assume that

$$\text{span}\{\mathbf{H}_0 \mathbf{g}_0, \dots, \mathbf{H}_0 \mathbf{g}_i\} = \text{span}\{\mathbf{H}_0 \mathbf{g}_0, \dots, (\mathbf{H}_0 \mathbf{A})^i \mathbf{H}_0 \mathbf{g}_0\}$$

for some $i < k$. Now,

$$\mathbf{g}_{i+1} = \mathbf{A}\mathbf{x}_{i+1} - \mathbf{b} = \mathbf{A}(\mathbf{x}_i + \mathbf{s}_i) - \mathbf{b} = \mathbf{A}\mathbf{s}_i + \mathbf{g}_i,$$

and from (2.16) and the induction hypothesis,

$$\mathbf{s}_i \in \text{span}\{\mathbf{H}_0\mathbf{g}_0, \dots, \mathbf{H}_0\mathbf{g}_i\} = \text{span}\{\mathbf{H}_0\mathbf{g}_0, \dots, (\mathbf{H}_0\mathbf{A})^i\mathbf{H}_0\mathbf{g}_0\},$$

which implies that $\mathbf{H}_0\mathbf{A}\mathbf{s}_i \in \text{span}\{(\mathbf{H}_0\mathbf{A})\mathbf{H}_0\mathbf{g}_0, \dots, (\mathbf{H}_0\mathbf{A})^{i+1}\mathbf{H}_0\mathbf{g}_0\}$. So,

$$\mathbf{H}_0\mathbf{g}_{i+1} \in \text{span}\{\mathbf{H}_0\mathbf{g}_0, \dots, (\mathbf{H}_0\mathbf{A})^{i+1}\mathbf{H}_0\mathbf{g}_0\}.$$

Hence, the search directions span the Krylov subspace. Since the search directions are conjugate (2.15) and span the Krylov subspace, the iterates are the same as those produced by conjugate gradients with preconditioner $\mathbf{H}_0$.

Since we produce the same iterates as the conjugate gradient method and the conjugate gradient method is well known to terminate within n iterations, we can conclude that this scheme terminates in at most n iterations. $\square$

Note that we require that $\mathbf{H}_k\mathbf{g}_k$ be nonzero whenever $\mathbf{g}_k$ is nonzero; this requirement is necessary since not all the methods produce positive definite updates and it is possible to construct an update that maps $\mathbf{g}_k$ to zero. If this were to happen, we would have a breakdown in the method.

The next corollary defines the role that the latest information ($\mathbf{s}_k$ and $\mathbf{y}_k$) plays in the formation of the k th $\mathbf{H}$ -update.

COROLLARY 2.4. *Suppose we have a method of the type described in Theorem 2.2 satisfying (2.17). Suppose further that at the k th iteration $\mathbf{P}_k$ is composed of p projections of the form in (2.2). Then at least one of the projections must have $\mathbf{u} = \sum_{i=0}^k \sigma_i \mathbf{y}_i$ with $\sigma_k \neq 0$. Furthermore, if $\mathbf{P}_k$ is a single projection ($p = 1$), then $\mathbf{v}$ must be of the form $\mathbf{v} = \rho_k \mathbf{s}_k + \rho_{k+1} \mathbf{s}_{k+1}$ with $\rho_k \neq 0$.*

Proof. Consider the case of $p = 1$. We have

$$\mathbf{P}_k = \mathbf{I} - \frac{\mathbf{u}\mathbf{v}^T}{\mathbf{v}^T\mathbf{u}},$$

where $\mathbf{u} \in \text{span}\{\mathbf{y}_0, \dots, \mathbf{y}_k\}$ and $\mathbf{v} \in \text{span}\{\mathbf{s}_0, \dots, \mathbf{s}_{k+1}\}$. We will assume that

$$\mathbf{u} = \sum_{i=0}^k \sigma_i \mathbf{y}_i \quad \text{and} \quad \mathbf{v} = \sum_{i=0}^{k+1} \rho_i \mathbf{s}_i$$

for some scalars σ_i and ρ_i . By (2.17), there exist $\mu_0, \dots, \mu_{k-1}$ such that

$$\mathbf{P}_k \mathbf{y}_k = \sum_{i=0}^{k-1} \mu_i \mathbf{y}_i.$$

Then

$$\mathbf{y}_k - \frac{\mathbf{v}^T \mathbf{y}_k}{\mathbf{v}^T \mathbf{u}} \mathbf{u} = \sum_{i=0}^{k-1} \mu_i \mathbf{y}_i,$$

and so

$$(2.20) \quad \frac{\mathbf{v}^T \mathbf{y}_k}{\mathbf{v}^T \mathbf{u}} \mathbf{u} = \mathbf{y}_k - \sum_{i=0}^{k-1} \mu_i \mathbf{y}_i.$$

From (2.15), the set $\{\mathbf{s}_0, \dots, \mathbf{s}_k\}$ is conjugate and thus linearly independent. Since we are working with a quadratic, $\mathbf{y}_i = \mathbf{A}\mathbf{s}_i$ for all i , and since $\mathbf{A}$ is symmetric positive definite, the set $\{\mathbf{y}_0, \dots, \mathbf{y}_k\}$ is also linearly independent. So the coefficient of the $\mathbf{y}_k$ on the left-hand side of (2.20) must match that on the right-hand side, thus

$$\frac{\mathbf{v}^T \mathbf{y}_k}{\mathbf{v}^T \mathbf{u}} \sigma_k = 1.$$

Hence, $\sigma_k \neq 0$ and $\mathbf{y}_k$ must make a nontrivial contribution to $\mathbf{P}_k$.

Next we will show that $\rho_0 = \rho_1 = \dots = \rho_{k-1} = 0$. Assume that j is between 0 and $k-1$. Then

$$\begin{aligned} \mathbf{P}_k \mathbf{y}_j &= \mathbf{y}_j - \frac{\mathbf{v}^T \mathbf{y}_j}{\mathbf{v}^T \mathbf{u}} \mathbf{u} \\ &= \mathbf{y}_j - \frac{\left(\sum_{i=1}^{k+1} \rho_i \mathbf{s}_i\right)^T \mathbf{y}_j}{\mathbf{v}^T \mathbf{u}} \mathbf{u} \\ &= \mathbf{y}_j - \frac{\sum_{i=1}^{k+1} \rho_i \mathbf{s}_i^T \mathbf{A} \mathbf{s}_j}{\mathbf{v}^T \mathbf{u}} \mathbf{u} \\ &= \mathbf{y}_j - \frac{\rho_j \mathbf{s}_j^T \mathbf{A} \mathbf{s}_j}{\mathbf{v}^T \mathbf{u}} \mathbf{u}. \end{aligned}$$

Now $\mathbf{s}_j \mathbf{A} \mathbf{s}_j$ is nonzero because $\mathbf{A}$ is positive definite. If ρ_j is nonzero, then the coefficient of $\mathbf{u}$ is nonzero, and so $\mathbf{y}_k$ must make a nontrivial contribution to $\mathbf{P}_k \mathbf{y}_j$, implying that $\mathbf{P}_k \mathbf{y}_j \notin \text{span}\{\mathbf{y}_0, \dots, \mathbf{y}_{k-1}\}$. This is a contradiction. Hence, $\rho_j = 0$.

To show that $\rho_k \neq 0$, consider $\mathbf{P}_k \mathbf{y}_k$. Suppose that $\rho_k = 0$. Then

$$\begin{aligned} \mathbf{v}^T \mathbf{y}_k &= \rho_{k+1} \mathbf{y}_k^T \mathbf{s}_{k+1} + \rho_k \mathbf{y}_k^T \mathbf{s}_k \\ &= \rho_{k+1} \mathbf{s}_k^T \mathbf{A} \mathbf{s}_{k+1} \\ &= 0, \end{aligned}$$

and so

$$\mathbf{P}_k \mathbf{y}_k = \mathbf{y}_k - \frac{\mathbf{v}^T \mathbf{y}_k}{\mathbf{v}^T \mathbf{u}} \mathbf{u} = \mathbf{y}_k.$$

This contradicts $\mathbf{P}_k \mathbf{y}_k \in \text{span}\{\mathbf{y}_0, \dots, \mathbf{y}_{k-1}\}$, so ρ_k must be nonzero.

Now we will discuss the $p > 1$ case. Label the $\mathbf{u}$ -components of the p projections as $\mathbf{u}_1$ through $\mathbf{u}_p$. Then

$$\mathbf{P}_k \mathbf{y}_k = \mathbf{y}_k + \sum_{i=1}^p \gamma_i \mathbf{u}_i$$

for some scalars γ_1 through γ_p . Furthermore, each $\mathbf{u}_i$ can be written as a linear combination of $\{\mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_k\}$, so

$$\mathbf{P}_k \mathbf{y}_k = \mathbf{y}_k + \sum_{i=1}^p \sum_{j=0}^k \gamma_i \sigma_{ij} \mathbf{y}_j$$

for some scalars σ_{10} through σ_{pk} . Since $\mathbf{P}_k \mathbf{y}_k \in \text{span}\{\mathbf{y}_0, \dots, \mathbf{y}_{k-1}\}$ and $\mathbf{y}_k \notin \text{span}\{\mathbf{y}_0, \dots, \mathbf{y}_{k-1}\}$, we must have

$$1 + \sum_{i=1}^p \gamma_i \sigma_{ik} = 0.$$

Thus σ_{ik} must be nonzero for some i , and we can conclude that at least one $\mathbf{u}_i$ must have a nontrivial contribution from $\mathbf{y}_k$. $\square$

2.2. Examples of methods that reproduce the conjugate gradient iterates. Here are some specific examples of methods that fit the general form, satisfy condition (2.17) of Theorem 2.2, and thus terminate in at most n iterations. The conjugate gradient, L-BFGS, and DFP examples are well-known results, but Corollary 2.5 is original.

Example 6. The conjugate gradient method with preconditioner $\mathbf{H}_0$ (see (2.4)) satisfies condition (2.17) of Theorem 2.2 since

$$\mathbf{P}_k \mathbf{y}_j = \left(\mathbf{I} - \frac{\mathbf{y}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{y}_k} \right) \mathbf{y}_j = \mathbf{0} \text{ for all } j = 0, \dots, k.$$

Example 7. L-BFGS (see (2.6)) satisfies condition (2.17) of Theorem 2.2 since

$$\mathbf{P}_k \mathbf{y}_j = \begin{cases} \mathbf{0} & \text{for } j = k - m_k + 1, \dots, k, \\ \mathbf{y}_j & \text{for } j = 0, \dots, k - m_k. \end{cases}$$

Example 8. DFP (with full memory) (see (2.8)) satisfies condition (2.17) of Theorem 2.2. Consider $\mathbf{P}_k$ in the full-memory case. We have

$$\mathbf{P}_k = \prod_{i=0}^k \left(\mathbf{I} - \frac{\mathbf{y}_i \mathbf{y}_i^T \mathbf{H}_i}{\mathbf{y}_i^T \mathbf{H}_i \mathbf{y}_i} \right).$$

For full-memory DFP, $\mathbf{H}_i \mathbf{y}_j = \mathbf{s}_j$ for $j = 0, \dots, i - 1$. Using this fact, one can easily verify that $\mathbf{P}_k \mathbf{y}_j = \mathbf{0}$ for $j = 0, \dots, k$. Therefore, full-memory DFP satisfies condition (2.17) of Theorem 2.2. The same reasoning does not apply to the limited-memory case, as we shall show in section 2.3.

The next corollary gives some ideas for other methods that are related to L-BFGS and terminate in at most n iterations on strictly convex quadratics.

COROLLARY 2.5. *The L-BFGS (2.5) method with exact line search will terminate in n iterations on an n -dimensional strictly convex quadratic function even if any combination of the following modifications is made to the update:*

1. *Vary the limited-memory constant, keeping $m_k \geq 1$.*
2. *Form the projections used in $\mathbf{V}_k$ from the most recent $(\mathbf{s}_k, \mathbf{y}_k)$ pair along with any set of $m - 1$ other pairs from $\{(\mathbf{s}_0, \mathbf{y}_0), \dots, (\mathbf{s}_{k-1}, \mathbf{y}_{k-1})\}$.*
3. *Form the projections used in $\mathbf{V}_k$ from the most recent $(\mathbf{s}_k, \mathbf{y}_k)$ pair along with any $m - 1$ other linear combinations of pairs from $\{(\mathbf{s}_0, \mathbf{y}_0), \dots, (\mathbf{s}_{k-1}, \mathbf{y}_{k-1})\}$.*
4. *Iteratively rescale $\mathbf{H}_0$.*

Proof. For each variant, we show that the method fits the general form in (2.1), satisfies condition (2.17) of Theorem 2.2, and hence terminates by Corollary 2.3:

1. Let $m > 0$ be any value which may change from iteration to iteration, and define

$$\mathbf{V}_{ik} = \prod_{j=i}^k \left(\mathbf{I} - \frac{\mathbf{y}_j \mathbf{s}_j^T}{\mathbf{s}_j^T \mathbf{y}_j} \right).$$

Choose

$$\begin{aligned}\gamma_k &= 1, \quad m_k = \min\{k+1, m\}, \\ \mathbf{P}_k &= \mathbf{Q}_k = \mathbf{V}_{k-m_k+1, k}, \quad \text{and} \\ \mathbf{w}_{ik} &= \mathbf{z}_{ik} = \frac{(\mathbf{V}_{k-m_k+i+1, k})^T (\mathbf{s}_{k-m_k+i})}{\sqrt{(\mathbf{s}_{k-m_k+i})^T (\mathbf{y}_{k-m_k+i})}}.\end{aligned}$$

These choices fit the general form. Furthermore,

$$\mathbf{P}_k \mathbf{y}_j = \begin{cases} \mathbf{0} & \text{if } j = k - m_k, k - m_k + 1, \dots, k, \\ \mathbf{y}_j & \text{if } j = 0, 1, \dots, k - m_k - 1, \end{cases}$$

so this variation satisfies condition (2.17) of Theorem 2.2.

2. This is a special case of the next variant.

3. At iteration k , let $(\hat{\mathbf{s}}_k^{(i)}, \hat{\mathbf{y}}_k^{(i)})$ denote the i th ($i = 1, \dots, m-1$) choice of any linear combination from the span of the set

$$\{(\mathbf{s}_0, \mathbf{y}_0), \dots, (\mathbf{s}_{k-1}, \mathbf{y}_{k-1})\},$$

and let $(\hat{\mathbf{s}}_k^{(m)}, \hat{\mathbf{y}}_k^{(m)}) = (\mathbf{s}_k, \mathbf{y}_k)$. Define

$$\mathbf{V}_{ik} = \prod_{j=i}^m \left(\mathbf{I} - \frac{(\hat{\mathbf{y}}_k^{(j)}) (\hat{\mathbf{s}}_k^{(j)})^T}{(\hat{\mathbf{s}}_k^{(j)})^T (\hat{\mathbf{y}}_k^{(j)})} \right).$$

Choose

$$\begin{aligned}\gamma_k &= 1, \quad m_k = \min\{k+1, m\}, \\ \mathbf{P}_k &= \mathbf{Q}_k = \mathbf{V}_{1, k}, \quad \text{and} \\ \mathbf{w}_{ik} &= \mathbf{z}_{ik} = \frac{(\mathbf{V}_{i+1, k})^T (\hat{\mathbf{s}}_k^{(i)})}{\sqrt{(\hat{\mathbf{s}}_k^{(i)})^T (\hat{\mathbf{y}}_k^{(i)})}}.\end{aligned}$$

These choices satisfy the general form (2.1). Furthermore,

$$\mathbf{P}_k \mathbf{y}_j = \begin{cases} \mathbf{0} & \text{if } \mathbf{y}_j = \mathbf{y}_k^{(i)} \text{ for some } i, \\ \mathbf{y}_j & \text{otherwise.} \end{cases}$$

Hence, this variation satisfies condition (2.17) of Theorem 2.2.

4. Let γ_k in (2.1) be the scaling constant and choose the other vectors and matrices as in L-BFGS (2.6).

Combinations of variants are left to the reader. $\square$

Remark 1. Part 3 of the previous corollary shows that the “accumulated step” method of Gill and Murray [11] terminates on quadratics.

Remark 2. Part 4 of the previous corollary shows that scaling does not affect termination in L-BFGS. In fact, for any method that fits the general form, it is easy to see that scaling will not affect termination on quadratics.

2.3. Examples of methods that do not reproduce the conjugate gradient iterates. We will discuss several methods that fit the general form given in (2.1) but do not satisfy the conditions of Theorem 2.2.

Example 9. Steepest descent (see (2.3)) does not satisfy condition (2.17) of Theorem 2.2 and thus does not produce conjugate search directions. This fact is well known; see, e.g., Luenberger [17].

Example 10. L-DFP (see (2.8)) with $m < n$ does not satisfy the condition on $\mathbf{P}_k$ (2.17) for all k , and so the method will not produce conjugate directions. This fact was previously unknown.

For example, suppose that we have a convex quadratic with

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

Using a limited-memory constant of $m = 1$ and exact arithmetic, it can be seen that the iteration does not terminate within the first 20 iterations of L-DFP with $\mathbf{H}_0 = \mathbf{I}$. The MAPLE notebook file used to compute this example is available on the World Wide Web [10].

Remark 3. Using the above example, we can easily see that no limited-memory Broyden family method except L-BFGS terminates within the first n iterations.

3. Update-skipping variations for Broyden family quasi-Newton algorithms. The previous section discussed limited-memory methods that behave like conjugate gradients on n -dimensional strictly convex quadratic functions. In this section, we are concerned with methods that skip some updates. The average computation cost per iteration is reduced and memory can be saved if the quasi-Newton matrix is stored implicitly. We establish conditions under which finite termination is preserved but delayed for the Broyden family.

3.1. Termination when updates are skipped. It was shown by Powell [27] that if we skip every other update and take *direct prediction steps* (i.e., steps of length one) in a Broyden family method, then the procedure will terminate in no more than $2n + 1$ iterations on an n -dimensional strictly convex quadratic function. An alternate proof of this result is given by Nazareth [22].

We will prove a related result. Suppose that we are doing *exact line searches* using a Broyden family quasi-Newton method on a strictly convex quadratic function and decide to “skip” p updates to $\mathbf{H}$ (i.e., choose $\mathbf{H}_{k+1} = \mathbf{H}_k$ on p occasions). Then the algorithm terminates in no more than $n + p$ iterations. The algorithm is somewhat more robust than using direct prediction steps; it does not matter which updates are skipped or if multiple updates are skipped in a row.

THEOREM 3.1. *Suppose that a Broyden family method using exact line searches is applied to an n -dimensional strictly convex quadratic function and p updates are skipped. Let*

$$J(k) = \{j \leq k : \text{the update at iteration } j \text{ is not skipped}\}.$$

Then for all $k = 0, 1, \dots$

$$(3.1) \quad \mathbf{g}_{k+1}^T \mathbf{s}_j = 0 \quad \text{for all } j \in J(k)$$

and

$$(3.2) \quad \mathbf{s}_{k+1}^T \mathbf{A} \mathbf{s}_j = 0 \quad \text{for all } j \in J(k).$$

Furthermore, the method terminates in at most $n + p$ iterations at the exact minimizer.

Proof. We will use induction on k to show (3.1) and

$$(3.3) \quad \mathbf{H}_{k+1}\mathbf{y}_j = \mathbf{s}_j \text{ for all } j \in J(k).$$

Then (3.2) follows easily since, for all $j \in J(k)$,

$$\begin{aligned} \mathbf{s}_{k+1}^T \mathbf{A} \mathbf{s}_j &= -\alpha_{k+1} \mathbf{g}_{k+1}^T \mathbf{H}_{k+1} \mathbf{y}_j \\ &= -\alpha_{k+1} \mathbf{g}_{k+1}^T \mathbf{s}_j \\ &= 0. \end{aligned}$$

Let k_0 be the least value of k such that $J(k)$ is nonempty; i.e., $J(k_0) = \{k_0\}$. Then $\mathbf{g}_{k_0+1}$ is orthogonal to $\mathbf{s}_{k_0}$ since line searches are exact, and $\mathbf{H}_{k_0+1}\mathbf{y}_{k_0} = \mathbf{s}_{k_0}$ since all members of the Broyden family satisfy the secant condition. Hence, the base case is true. Now assume that (3.1) and (3.3) hold for all values of $k = 0, 1, \dots, \hat{k} - 1$. We will show that they also hold for $k = \hat{k}$.

Case 1. Suppose that $\hat{k} \notin J(\hat{k})$. Then $\mathbf{H}_{\hat{k}+1} = \mathbf{H}_{\hat{k}}$ and $J(\hat{k} - 1) = J(\hat{k})$, so, for any $j \in J(\hat{k})$,

$$(3.4) \quad \mathbf{g}_{\hat{k}+1}^T \mathbf{s}_j = 0$$

and

$$\mathbf{H}_{\hat{k}+1}\mathbf{y}_j = \mathbf{H}_{\hat{k}}\mathbf{y}_j = \mathbf{s}_j.$$

Case 2. Suppose that $\hat{k} \in J(\hat{k})$. Then $\mathbf{H}_{\hat{k}+1}$ satisfies the secant condition and $J(\hat{k}) = J(\hat{k} - 1) \cup \{\hat{k}\}$. Now $\mathbf{g}_{\hat{k}+1}$ is orthogonal to $\mathbf{s}_k$ since the line searches are exact, and it is orthogonal to the older $\mathbf{s}_j$ by the argument in (3.4). The secant condition guarantees that $\mathbf{H}_{\hat{k}+1}\mathbf{y}_{\hat{k}} = \mathbf{s}_{\hat{k}}$, and, for $j \in J(\hat{k})$ but $j \neq \hat{k}$, we have

$$\begin{aligned} \mathbf{H}_{\hat{k}+1}\mathbf{y}_j &= \mathbf{H}_{\hat{k}}\mathbf{y}_j + \frac{\mathbf{s}_{\hat{k}}\mathbf{s}_{\hat{k}}^T}{\mathbf{s}_{\hat{k}}^T\mathbf{y}_{\hat{k}}}\mathbf{y}_j - \frac{\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}}\mathbf{y}_{\hat{k}}^T\mathbf{H}_{\hat{k}}}{\mathbf{y}_{\hat{k}}^T\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}}}\mathbf{y}_j \\ &\quad + \phi(\mathbf{y}_{\hat{k}}^T\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}})\left(\frac{\mathbf{s}_{\hat{k}}}{\mathbf{s}_{\hat{k}}^T\mathbf{y}_{\hat{k}}} - \frac{\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}}}{\mathbf{y}_{\hat{k}}^T\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}}}\right)\left(\frac{\mathbf{s}_{\hat{k}}}{\mathbf{s}_{\hat{k}}^T\mathbf{y}_{\hat{k}}} - \frac{\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}}}{\mathbf{y}_{\hat{k}}^T\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}}}\right)^T\mathbf{y}_j \\ &= \mathbf{s}_j + \frac{\mathbf{s}_{\hat{k}}^T\mathbf{A}\mathbf{s}_j}{\mathbf{s}_{\hat{k}}^T\mathbf{y}_{\hat{k}}}\mathbf{s}_{\hat{k}} - \frac{\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}}\mathbf{y}_{\hat{k}}^T\mathbf{s}_j}{\mathbf{y}_{\hat{k}}^T\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}}} \\ &\quad + \phi(\mathbf{y}_{\hat{k}}^T\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}})\left(\frac{\mathbf{s}_{\hat{k}}}{\mathbf{s}_{\hat{k}}^T\mathbf{y}_{\hat{k}}} - \frac{\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}}}{\mathbf{y}_{\hat{k}}^T\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}}}\right)\left(\frac{\mathbf{s}_{\hat{k}}^T\mathbf{A}\mathbf{s}_j}{\mathbf{s}_{\hat{k}}^T\mathbf{y}_{\hat{k}}} - \frac{\mathbf{y}_{\hat{k}}^T\mathbf{s}_j}{\mathbf{y}_{\hat{k}}^T\mathbf{H}_{\hat{k}}\mathbf{y}_{\hat{k}}}\right) \\ &= \mathbf{s}_j. \end{aligned}$$

In either case, the induction result follows.

Suppose that we skip p updates. Then the set $J(n - 1 + p)$ has cardinality n . Without loss of generality, assume that the set $\{\mathbf{s}_i\}_{i \in J(n-1+p)}$ has no zero elements. From (3.2), the vectors are linearly independent. By (3.1),

$$\mathbf{g}_{n+p}^T \mathbf{s}_j = 0 \quad \text{for all } j \in J(n - 1 + p),$$

and so $\mathbf{g}_{n+p}$ must be zero. This implies that $\mathbf{x}_{n+p}$ is the exact minimizer of f . $\square$

3.2. Loss of termination for update skipping with limited-memory. Unfortunately, updates that use both limited-memory and repeated update-skipping do not produce conjugate search directions for n -dimensional strictly convex quadratics, and the termination property is lost. We will show a simple example.

Example 11. Suppose that we have a convex quadratic with

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

We apply L-BFGS with limited-memory constant $m = 1$ and $\mathbf{H}_0 = \mathbf{I}$ and skip every other update to $\mathbf{H}$. Using exact arithmetic in MAPLE, we observe that the process does not terminate even after 100 iterations [10]. Note that, according to Corollary 2.4, we would still be guaranteed termination if we used the most recent information in each update.

4. Experimental results. Thus far we have only given results for convex quadratic functions. While termination on quadratics is beautiful in theory, it does not necessarily yield insight into how these methods will do in practice. In this section, we develop and compare some of the new methods. We describe the collection of test problems in section 4.2. Complete numerical results, many graphs of the numerical results, and the original FORTRAN code are available [10].

4.1. Motivation. We will not present any new results relating to convergence of these algorithms on general functions; however, many of these can be shown to converge using the convergence analysis presented in section 7 of [16]. In [16], Liu and Nocedal show that an L-BFGS method implemented with a line search that satisfies the strong Wolfe conditions (see section 4.3 for a definition) is R-linearly convergent on a convex function that satisfies a few modest conditions.

4.2. Test problems. For our test problems, we used the CUTE by Bongartz, Conn, Gould, and Toint. The package is documented in [3] and can be obtained via the World Wide Web [2] or via ftp [1]. The package contains a large collection of test problems as well as the interfaces necessary for using the problems. We chose a collection of 22 unconstrained problems. The problems ranged in size from 10 to 10,000 variables, but each took L-BFGS with limited-memory constant $m = 5$ at least 60 iterations to solve. Table 4.1 enumerates the problems, giving the SIF file name, the dimension (n), and a description for each problem. The CUTE package also provides a starting point (x_0) for each problem.

4.3. Test environment. We used FORTRAN77 code on an SGI Indigo² to run the algorithms, with FORTRAN BLAS routines from NETLIB. We used the compiler's default optimization level.

Figure 2.1 outlines the general quasi-Newton implementation that we followed. For the line search, we use the routines `cvsrch` and `cstep` written by Jorge J. Moré and David Thuente from a 1983 version of MINPACK. This line search routine finds an α that meets the strong Wolfe conditions

$$(4.1) \quad f(\mathbf{x} + \alpha \mathbf{d}) \leq f(\mathbf{x}) + \omega_1 \alpha \mathbf{g}(\mathbf{x})^T \mathbf{d},$$

$$(4.2) \quad |\mathbf{g}(\mathbf{x} + \alpha \mathbf{d})^T \mathbf{d}| \leq \omega_2 |\mathbf{g}(\mathbf{x})^T \mathbf{d}|;$$

see, e.g., Nocedal [24]. We used $\omega_1 = 1.0 \times 10^{-4}$ and $\omega_2 = 0.9$. Except for the first iteration, we always attempt a step length of 1.0 first and only use an alternate value

TABLE 4.1

Test problem collection. Each problem was chosen from the CUTE package.

No.	SIF name	n	Description and reference
1	EXTROSNB	10	Extended Rosenbrock function (nonseparable version) [30, Problem 10].
2	WATSONS	31	Watson problem [18, Problem 20].
3	TOINTGOR	50	Toint's operations research problem [29].
4	TOINTPSP	50	Toint's PSP operations research problem [29].
5	CHNROSNB	50	Chained Rosenbrock function [29].
6	ERRINROS	50	Nonlinear problem similar to CHNROSNB [3].
7	FLETCHBV	100	Fletcher's boundary value problem [9, Problem 1].
8	FLETCHCR	100	Fletcher's chained Rosenbrock function [9, Problem 2].
9	PENALTY2	100	Second penalty problem [18, Problem 24].
10	GENROSE	500	Generalized Rosenbrock function [19, Problem 5].
11	BDQRTIC	1000	Quartic with a banded Hessian with bandwidth = 9 [6, Problem 61].
12	BROYDN7D	1000	Seven diagonal variant of the Broyden tridiagonal system with a band away from diagonal [29].
13	PENALTY1	1000	First penalty problem [18, Problem 23].
14	POWER	1000	Power problem by Oren [26].
15	MSQRTALS	1024	The dense matrix square root problem by Nocedal and Liu (case 0) seen as a nonlinear equation problem [4, Problem 204].
16	MSQRTBLS	1025	The dense matrix square root problem by Nocedal and Liu (case 1) seen as a nonlinear equation problem [4, Problem 201].
17	CRAGGLVY	5000	Extended Cragg and Levy problem [30, Problem 32].
18	NONDQUAR	10000	Nondiagonal quartic test problem [6, Problem 57].
19	POWELLSG	10000	Extended Powell singular function [18, Problem 13].
20	SINQUAD	10000	Another function with nontrivial groups and repetitious elements [13].
21	SPMSRTLS	10000	Liu and Nocedal tridiagonal matrix square root problem [4, Problem 151].
22	TRIDIA	10000	Shanno's TRIDIA quadratic tridiagonal problem [30, Problem 8].

if 1.0 does not satisfy the Wolfe conditions. In the first iteration, we initially try a step length equal to $\|\mathbf{g}_0\|^{-1}$. The remaining line search parameters are detailed in Table 4.2.

We generate the matrix $\mathbf{H}_k$ by either the limited-memory update or one of the variations described in section 4.4, storing the matrix implicitly in order to save both memory and computation time.

TABLE 4.2

Line search parameters.

Variable	Value	Description
STP	1.0	Step length to try first.
XTOL	1.0×10^{-15}	Relative width of interval of uncertainty.
STPMIN	1.0×10^{-15}	Minimum step length.
STPMAX	1.0×10^{15}	Maximum step length.
MAXFEV	20	Maximum number of function evaluations.

We terminate at iteration k if any of the following conditions is met:

1. the iterate satisfies

$$\frac{\|\mathbf{g}_k\|}{\|\mathbf{x}_k\|} < 1.0 \times 10^{-5};$$

2. the line search fails to satisfy both (4.1) and (4.2); or
3. the number of iterations exceeds 3000.

We say that the iterates have converged if the first condition is satisfied. Otherwise, the method has failed.

4.4. L-BFGS and its variations. We tried a number of variations to the standard L-BFGS algorithm. L-BFGS and these variations are described in this subsection and summarized in Table 4.3.

TABLE 4.3
Description of numerical algorithms.

No.	Reference	Brief Description
0	Section 4.4.1	L-BFGS with no options.
1	Section 4.4.2, Variation 4	Allow m to vary iteratively, basing the choice of m on $\ \mathbf{g}\ /\ \mathbf{x}\ $ and allowing m to decrease.
2	Section 4.4.3	Dispose of old information if the step length is greater than one.
3	Section 4.4.4, Variation 1	Back up if the current iteration is odd.
4	Section 4.4.4, Variation 3	Back up if a step length of 1.0 was used in the last iteration.
5	Section 4.4.5, Variation 2	Merge the 2nd and 3rd most recent $(\mathbf{s}, \mathbf{y})$ pairs if the corresponding step lengths were 1 and neither pair is the result of a previous merge.
6	Section 4.4.6, Variation 1	Skip update on odd iterations.
7	Algorithm 2 and Algorithm 4	Dispose of old information and back up on the next iteration if the step length is greater than one.
8	Algorithm 6 and Algorithm 1	Merge if we did not do a merge in the last iteration and there are at least two old $\mathbf{s}$ vectors to merge, and allow m to vary iteratively, basing the choice of m on $\ \mathbf{g}\ /\ \mathbf{x}\ $ and allowing m to decrease.

4.4.1. L-BFGS: Algorithm 0. The L-BFGS update is given in (2.5) and described fully by Byrd, Nocedal, and Schnabel [5].

The storage costs are $O(mn)$, rather than n^2 required by BFGS. The computation of $\mathbf{H}\mathbf{g}$ takes at most $O(mn)$ operations rather than $O(n^2)$.

We are using L-BFGS as our basis for comparison. For information on the performance of L-BFGS, see Liu and Nocedal [16] and Nash and Nocedal [20].

4.4.2. Varying m iteratively: Algorithm 1. In typical implementations of L-BFGS, m is fixed throughout the iterations; once m updates have accumulated, m updates are always used. We considered the possibility of varying m iteratively, preserving finite termination on convex quadratics. Using an argument similar to that presented in [16], we can also prove that this algorithm has a linear rate of convergence on a convex function that satisfies a few modest conditions.

We scaled m in relation to the size of $\|\mathbf{g}\|/\max\{1, \|\mathbf{x}\|\}$. Let m_k be the number of iterates saved at the k th iteration, with $m_0 = 1$. Here, think of m as the maximum allowable value of m_k . Let the convergence test be given by $\|\mathbf{g}_k\|/\max\{1, \|\mathbf{x}_k\|\} < \epsilon$.

TABLE 4.4

The number of failures of the algorithms on the 22 test problems. An algorithm is said to have “failed” on a particular problem if a line search fails or the maximum allowable number of iterations (3000 in our case) is exceeded.

Alg. No.	$m = 5$	$m = 10$	$m = 15$	$m = 50$
0	1	0	0	1
1	1	0	0	1
2	0	0	0	0
3	1	0	0	1
4	0	0	0	0
5	1	0	0	1
6	12	12	12	12
7	0	0	0	0
8	3	1	0	1

TABLE 4.5

Function evaluations comparison. The first number in each entry is the number of times the algorithm did as well as or better than normal L-BFGS in terms of function evaluations. The second number is the total number of problems solved by at least one of the two methods (the algorithm and/or L-BFGS).

Alg. No.	$m = 5$	$m = 10$	$m = 15$	$m = 50$
1	12/21	17/22	15/22	16/21
2	19/22	20/22	20/22	21/22
3	21/21	22/22	22/22	21/21
4	12/22	14/22	12/22	15/22
5	3/22	4/22	4/22	4/22
6	1/21	1/22	1/22	1/21
7	12/22	13/22	12/22	14/22
8	1/22	2/22	4/22	4/22

Then the formula for m_k at iteration k is

$$m_k = \min \left\{ m_{k-1} + 1, \left\lceil (m-1) \frac{\log \frac{\|\mathbf{g}_k\|}{\max\{1, \|\mathbf{x}_k\|\}} - \log \frac{\|\mathbf{g}_0\|}{\max\{1, \|\mathbf{x}_0\|\}}}{\log 100\epsilon - \log \frac{\|\mathbf{g}_0\|}{\max\{1, \|\mathbf{x}_0\|\}}} \right\rceil + 1 \right\}.$$

We used four values of m : 5, 10, 15, and 50. The results are summarized in Tables 4.4–4.8. More extensive results are given in [10].

Table 4.4 shows that this algorithm had the same number of failures as L-BFGS.

Table 4.5 compares the algorithm to L-BFGS in terms of function evaluations. The number of times that the algorithm used *as few or fewer* function evaluations than L-BFGS is listed relative to the total number of admissible problems. Problems are admissible if either of the two methods solved it. This algorithm used as few or fewer function evaluations than L-BFGS for over half the test problems.

Table 4.6 compares this algorithm to L-BFGS in terms of time. The entries are similar to those in Table 4.5. Observe that Algorithm 1 did very well in terms of time, doing as well or better than L-BFGS on approximately 80% of the problems.

For each problem, we computed the ratio of the number of function evaluations for the algorithm to the number of function evaluations for L-BFGS. Table 4.7 lists the means of these ratios. A mean below 1.0 implies that the algorithm does better than L-BFGS on average. The average is better for the first algorithm in 2 out of 4 cases. Observe, however, that all the means are close to one.

TABLE 4.6

Time comparison. The first number in each entry is the number of times the algorithm did as well as or better than normal L-BFGS in terms of time. The second number is the total number of problems solved by at least one of the two methods (the algorithm and/or L-BFGS).

Alg. No.	$m = 5$	$m = 10$	$m = 15$	$m = 50$
1	17/21	18/22	20/22	18/21
2	15/22	13/22	14/22	15/22
3	16/21	19/22	15/22	15/21
4	11/22	7/22	6/22	5/22
5	5/22	10/22	13/22	17/22
6	1/21	1/22	1/22	2/21
7	11/22	8/22	5/22	4/22
8	9/22	16/22	16/22	18/22

TABLE 4.7

Mean function evaluations ratios for each algorithm compared to L-BFGS. Problems for which either method failed are not used in this mean.

Alg. No.	$m = 5$	$m = 10$	$m = 15$	$m = 50$
1	0.998	1.297	0.970	1.000
2	1.021	0.971	1.005	1.010
3	1.000	1.000	1.000	1.000
4	0.991	1.677	1.507	0.891
5	1.137	1.178	1.244	1.373
6	8.227	8.666	9.073	9.308
7	0.981	1.023	0.924	0.918
8	1.406	1.161	1.178	1.394

This algorithm tends to save fewer vectors than L-BFGS since m_k is typically less than m and so less work is done computing $\mathbf{H}_k \mathbf{g}_k$. Table 4.8 gives the mean of the ratios of time to solve for each value of m in each algorithm. Note that most of the ratios are far below one.

4.4.3. Disposing of old information: Algorithm 2. We may decide that we should stop using outdated information. For example, we may choose to keep only the most recent information whenever we take a big step, since the old information may not be relevant to the new neighborhood. We use the following test: If the last step length was bigger than 1, discard all but the most recent $\mathbf{s}$ and $\mathbf{y}$ pair.

The algorithm performed nearly the same as L-BFGS. There was substantial deviation on only one or two problems for each value of m , and this seemed evenly divided in terms of better and worse. From Table 4.4, we see that this algorithm successfully converged on every problem. Table 4.5 shows that it almost always did as well or better than L-BFGS in terms of function evaluations. However, Table 4.7 shows that the differences were minor. The algorithm generally took less time than L-BFGS (Table 4.6), but again, considering the mean ratios of time (Table 4.8), the differences were minor.

4.4.4. Backing up in the update to $\mathbf{H}$: Algorithms 3-4. As discussed in section 2.2, if we always use the most recent $\mathbf{s}$ and $\mathbf{y}$ in the update, we preserve quadratic termination regardless of which older values of $\mathbf{s}$ and $\mathbf{y}$ we use.

Using this idea, we created some algorithms. Under certain conditions, we discard the next most recent values of $\mathbf{s}$ and $\mathbf{y}$ although we still use the *most* recent $\mathbf{s}$ and $\mathbf{y}$ vectors and any other vectors that have been saved from previous iterations. We call

TABLE 4.8

Mean time ratios for each algorithm compared to L-BFGS. Problems for which either method failed are not used in this mean.

Alg. No.	$m = 5$	$m = 10$	$m = 15$	$m = 50$
1	0.907	1.119	0.823	0.856
2	1.041	0.969	0.993	1.004
3	1.007	0.983	0.977	0.995
4	1.057	1.421	1.426	1.425
5	1.083	1.082	0.983	0.960
6	5.008	4.046	3.527	2.646
7	1.053	1.166	1.089	1.399
8	1.258	0.927	0.859	0.974

this “backing up” because it is as if we back up over the next most recent update. These algorithms used the following tests to trigger backing up:

1. The current iteration is odd.
2. A step length of 1.0 was used in the last iteration.

Algorithm 3 did not fail at all. See Table 4.4 for more information.

Backing up on odd iterations (Algorithm 3) seemed to have almost no effect on the number of function evaluations (Table 4.7) and little effect on the time (Table 4.8).

In Algorithm 4, we back up if the previous step length was one. This wipes out the data from the previous iteration after it has been used in one update. It is an improvement over L-BFGS in terms of function evaluations; in fact, this algorithm has the best function evaluation ratio for the $m = 50$ case (Table 4.7). Unfortunately, this algorithm did not compete with L-BFGS in terms of time (Table 4.8).

4.4.5. Merging $\mathbf{s}$ and $\mathbf{y}$ information in the update: Algorithm 5. Yet another idea is to “merge” $\mathbf{s}$ data to use less storage and computation time. By merging, we mean forming some linear combination of various $\mathbf{s}$ vectors. The $\mathbf{y}$ vectors would be merged correspondingly. Corollary 2.5 shows that as long as the most recent $\mathbf{s}$ and $\mathbf{y}$ are used without merge, old $\mathbf{s}$ vectors may be replaced by any linear combination of the old $\mathbf{s}$ vectors in L-BFGS.

We used this idea in the following way: we merged the 2nd and 3rd most recent $(\mathbf{s}, \mathbf{y})$ pairs if the corresponding step lengths were 1 and neither pair was the result of a previous merge. A merge is accomplished by adding the two pairs together and replacing the two pairs with the single “sum” pair.

Algorithm 5 is better than L-BFGS in terms of time, especially for the larger values of m (Tables 4.6 and 4.8). Unfortunately, this reflects only a saving in the amount of linear algebra required. The number of function evaluations generally is larger for this algorithm than L-BFGS (Tables 4.5 and 4.7).

4.4.6. Skipping updates to $\mathbf{H}$: Algorithm 6. If every other update to $\mathbf{H}$ is skipped and a step length of one is always chosen, BFGS will terminate in $2n$ iterations on a strictly convex quadratic function. The same holds true when doing an exact line search. (See section 3.) Unfortunately, neither property holds in the limited-memory case. We will, however, try an algorithm motivated by this idea.

Skipping on odd updates (Algorithm 6) did extremely well for every value of m only on problem 1. Otherwise, it did very badly.

4.4.7. Combined methods: Algorithms 7–8. We did some experimentation with combinations of methods described in the previous sections.

In Algorithm 7, we combined Algorithms 2 and 4; we dispose of old information and back up on the next iterations if the step length is greater than one. Essentially we are assuming that we have stepped out of the region being modeled by the quasi-Newton matrix if we take a long step, and we should thus rid the quasi-Newton matrix of that information. This algorithm did well in terms of function evaluations, having mean ratios of less than one for three values of m (Table 4.7), but it did not do as well in terms of time.

In Algorithm 8, we combined merging (Algorithm 5) and varying m (Algorithm 1). This algorithm did well in terms of time for larger m (Table 4.8) but not in terms of function evaluations (Table 4.7).

5. Conclusions. There is a spectrum of quasi-Newton methods, ranging from those that require the storage of an $n \times n$ approximate Hessian (e.g., the Broyden family) to those that require only the storage of a few vectors (e.g., conjugate gradients). Limited-memory quasi-Newton methods fall in between these extremes in terms of performance and storage. There are other methods that fall into the middle ground; for example, conjugate gradient methods such as those proposed by Shanno [28] and Nazareth [21], the truncated-Newton method [25, 7], and the partitioned quasi-Newton method [14].

We have characterized which limited-memory quasi-Newton methods fitting a general form (2.1) have the property of producing conjugate search directions on convex quadratics. We have shown that L-BFGS is the only Broyden family member that has a limited-memory analog with this property. We also considered update-skipping, something that may seem attractive in a parallel environment. We show that update skipping on quadratic problems is acceptable for full-memory Broyden family members in that it only delays termination, but that we lose the property of finite termination if we both limit memory and skip updates.

We have also introduced some simple-to-implement modifications of the standard L-BFGS algorithm that seem to behave well on some practical problems.

REFERENCES

- [1] I. BONGARTZ, A. R. CONN, N. GOULD, AND P. L. TOINT, CUTE ftp site, <ftp://thales.math.fundp.ac.be/pub/cute>.
- [2] I. BONGARTZ, A. R. CONN, N. GOULD, AND P. L. TOINT, CUTE home page, <http://www.dci.clrc.ac.uk/Activity.asp?CUTE>.
- [3] I. BONGARTZ, A. R. CONN, N. GOULD, AND P. L. TOINT, *CUTE: Constrained and unconstrained testing environment*, ACM Trans. Math. Software, 21 (1995), pp. 123–160.
- [4] A. BUCKLEY, *Test functions for unconstrained minimization*, Tech. Report TR 1989CS-3, Mathematics, Statistics and Computing Centre, Dalhousie University, Halifax (CDN), 1989.
- [5] R. H. BYRD, J. NOCEDAL, AND R. B. SCHNABEL, *Representations of quasi-Newton matrices and their use in limited memory methods*, Math. Programming, 63 (1994), pp. 129–156.
- [6] A. CONN, N. GOULD, M. LESCENIER, AND P. TOINT, *Performance of a multifrontal scheme for partially separable optimization*, Tech. Report 88/4, Department of Mathematics, FUNDP, Namur, Belgium, 1988. Cited in [1, 2].
- [7] R. S. DEMBO AND T. STEihaug, *Truncated-Newton algorithms for large-scale unconstrained optimization*, Math. Programming, 26 (1983), pp. 190–212.
- [8] J. DENNIS AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Series in Computational Mathematics, Prentice-Hall, Englewood Cliffs, NJ, 1983. Reprinted by Society for Industrial and Applied Mathematics, Philadelphia, PA, 1996.
- [9] R. FLETCHER, *An optimal positive definite update for sparse Hessian matrices*, Numerical Analysis NA/145, Technical report, University of Dundee, 1992. Cited in [1, 2].

- [10] T. GIBSON, D. O'LEARY, AND L. NAZARETH, L-BFGS with Variations home page, <http://www.cs.umd.edu/users/oleary/LBFGS/index.html> (1996).
- [11] P. E. GILL AND W. MURRAY, *Conjugate-gradient methods for large-scale nonlinear optimization*, Tech. Report SOL 79-15, Systems Optimization Laboratory, Department of Operations Research, Stanford University, Stanford, CA, 1979.
- [12] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, 2nd ed., The Johns Hopkins University Press, Baltimore, 1989.
- [13] N. GOULD, Private communication to authors of [3], 1989. Cited in [1, 2].
- [14] A. GRIEWANK AND P. L. TOINT, *Partitioned variable metric updates for large structured optimization problems*, Numer. Math., 39 (1982), pp. 119–137.
- [15] H. KHALFAN, R. BYRD, AND R. SCHNABEL, *A theoretical and experimental study of the symmetric rank one update*, Tech. Report CU-CS-489-90, Department of Computer Science, University of Colorado at Boulder, 1990.
- [16] D. C. LIU AND J. NOCEDAL, *On the limited memory BFGS method for large scale optimization*, Math. Programming, 45 (1989), pp. 503–528.
- [17] D. G. LUENBERGER, *Linear and Nonlinear Programming*, 2nd ed., Addison-Wesley, Reading, MA, 1984.
- [18] J. J. MORÉ, B. S. GARBOW, AND K. E. HILLSTROM, *Testing unconstrained optimization software*, ACM Trans. Math. Software, 7 (1981), pp. 17–41.
- [19] S. NASH, *Newton-type minimization via the Lanczos process*, SIAM J. Numer. Anal., 21 (1984), pp. 770–788.
- [20] S. G. NASH AND J. NOCEDAL, *A numerical study of the limited memory BFGS method and the truncated-Newton method for large scale optimization*, SIAM J. Optim., 1 (1991), pp. 358–372.
- [21] L. NAZARETH, *A relationship between BFGS and conjugate gradient algorithms and its implications for new algorithms*, SIAM J. Numer. Anal., 16 (1979), pp. 794–800.
- [22] L. NAZARETH, *On the BFGS Method*, unpublished manuscript, University of California at Berkeley, Berkeley, CA, 1981.
- [23] J. NOCEDAL, *Updating quasi-Newton matrices with limited storage*, Math. Comp., 35 (1980), pp. 773–782.
- [24] J. NOCEDAL, *Theory of algorithms for unconstrained optimization*, in Acta Numerica (1991), Cambridge University Press, London, 1992, pp. 199–242.
- [25] D. P. O'LEARY, *A discrete Newton algorithm for minimizing a function of many variables*, Math. Programming, 23 (1982), pp. 20–33.
- [26] S. OREN, *Self-scaling variable metric algorithms, Part II: Implementation and experiments*, Management Science, 20 (1974), pp. 863–874. Cited in [1, 2].
- [27] M. J. D. POWELL, *Quadratic termination properties of minimization algorithms I: Statement and discussion of results*, J. Inst. Math. Appl., 10 (1972), pp. 333–342.
- [28] D. F. SHANNO, *Conjugate gradient methods with inexact line searches*, Math. Oper. Res., 3 (1978), pp. 244–256.
- [29] P. TOINT, *Some numerical results using a sparse matrix updating formula in unconstrained optimization*, Math. Comput., 32 (1978), pp. 839–852.
- [30] P. TOINT, *Test problems for partially separable optimization and results for the routine PSP-MIN*, Tech. Report 83/4, Department of Mathematics, FUNDP, Namur, Belgium, 1983. Cited in [1, 2].

OPTIMAL TRUSS DESIGN BY INTERIOR-POINT METHODS*

FLORIAN JARRE[†], MICHAL KOČVARA[‡], AND JOCHEM ZOWE[‡]

Abstract. This article presents a primal-dual predictor-corrector interior-point method for solving quadratically constrained convex optimization problems that arise from truss design problems. We investigate certain special features of the problem, discuss fundamental differences of interior-point methods for linearly and nonlinearly constrained problems, extend Mehrotra's predictor-corrector strategy to nonlinear programs, and establish convergence of a long step method. Numerical experiments on large scale problems illustrate the surprising efficiency of the method.

Key words. interior-point methods, truss topology design, convex programming

AMS subject classifications. 90C30, 73K40

PII. S1052623496297097

1. Introduction.

1.1. The mechanical-mathematical model. In this section we briefly sketch the engineering problem which we propose to solve by interior-point methods. A more detailed introduction can be found in [14]. The reader who is not interested in the physical background can switch immediately to the final mathematical formulation (1.8).

We consider the problem of finding the stiffest truss (pin-jointed framework) with respect to a given load. This problem is studied in the so-called *ground-structure* framework [8], where we start from a dense mesh of N nodal points in $\mathbb{R}^{dim}$, $dim = 2$ or 3, and each two of these nodes can be connected by a bar. Thus we work with N tentative nodes and with up to $m = N(N - 1)/2$ potential bars. Our optimization problem is to select from this rich network the most profitable set of nodes and bars able to carry a given load f (the thin arrows at the right of each layout in Figure 1.1).

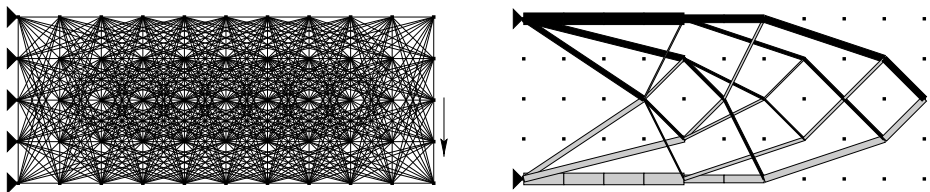


FIG. 1.1. Initial layout and optimal truss with vertical load on the right-hand side.

The design variables (controls) in the model are the bar volumes $t_i \geq 0, i = 1, \dots, m$, which are aggregated in a vector t . Further let x denote the vector of nodal

*Received by the editors January 2, 1996; accepted for publication (in revised form) December 11, 1997; published electronically September 23, 1998. This research was partly supported by project 03ZO7BAY of the Federal Department of Education, Science, Research and Technology (BMBF), Germany, and grant A1075707 of the Czech Academy of Sciences.

<http://www.siam.org/journals/siopt/8-4/29709.html>

[†] and Institut für Angewandte Mathematik und Statistik, Universität Würzburg, Am Hubland, D-97074 Würzburg, Germany (jarre@mathematik.uni-wuerzburg.de).

[‡]Institut für Angewandte Mathematik, Universität Erlangen, Martensstr. 3, D-91058 Erlangen, Germany (kocvara@am.uni-erlangen.de, zowe@am.uni-erlangen.de).

displacements of the system under load. We assume that there are p components with prescribed fixed displacement (that are “nailed” to a solid wall, for example). These components can be omitted from the problem formulation and thus x is of dimension $n = \dim \cdot N - p$.

Under the assumption of linear elastic material behavior, the potential energy of such a truss structure, which is subject to an external nodal force vector $f \in \mathbb{R}^n$, is given by

$$(1.1) \quad \Pi(t, x) = \frac{1}{2} x^T \left(\sum_{i=1}^m t_i A_i \right) x - f^T x;$$

here

$$A_i = b_i b_i^T \quad \text{with} \quad b_i = \frac{\sqrt{E_i}}{l_i} \gamma_i, \quad i = 1, 2, \dots, m,$$

is the symmetric positive semidefinite $n \times n$ dyadic stiffness matrix of the i th bar, E_i is a material constant (Young’s modulus), l_i is the length of the bar, and $\gamma_i \in \mathbb{R}^n$ is the vector of direction cosines such that $\gamma_i^T x$ measures the bar elongation.

Throughout we assume that

$$(1.2) \quad b_1, \dots, b_m \text{ span the whole space } \mathbb{R}^n.$$

It is straightforward to show that under this assumption

$$(1.3) \quad \sum_{i=1}^m t_i A_i \text{ is positive definite whenever } t_i > 0 \text{ for } i = 1, \dots, m.$$

The x that minimizes the potential energy $\Pi(t, \cdot)$ determines a state of equilibrium in which the inner and outer forces balance each other. The goal of the designer is to find some t under the (normalized) volume constraint $\sum_{i=1}^m t_i = 1$, for which this minimum is as close to zero as possible. Since $\min_{x \in \mathbb{R}^n} \Pi(t, x)$ is equal to or less than zero, the design problem becomes

$$(1.4) \quad \max_{t \in \mathbb{R}^m} \min_{x \in \mathbb{R}^n} \Pi(t, x) \quad \text{subject to} \quad t_i \geq 0, \quad i = 1, \dots, m, \\ \sum_{i=1}^m t_i = 1.$$

In [14] it is shown that “max” and “min” in (1.4) and the following formulas are justified and optimal solutions exist.

Another often used formulation of the design problem is in terms of “minimization of compliance” (“maximization of global stiffness”):

$$(1.5) \quad \min_{t \in \mathbb{R}^m, x \in \mathbb{R}^n} \frac{1}{2} f^T x \quad \text{subject to} \quad \sum_{i=1}^m t_i A_i x = f, \\ t_i \geq 0, \quad i = 1, \dots, m, \\ \sum_{i=1}^m t_i = 1.$$

Formulations (1.4) and (1.5) are indeed equivalent up to a factor of -1 ; i.e., they yield the same optimizers (x, t) , and the minimal values differ only by a factor of

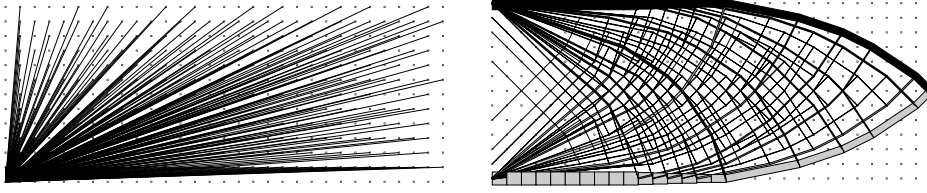


FIG. 1.2. Refinement of example from Figure 1.1.

–1. To realize this we note that only t with $f \in \text{range}(\sum_{i=1}^m t_i A_i)$ play a role in the minimization (1.5); such t exist due to (1.2) and (1.3). Now fix any feasible t and note that

$$f \in \text{range}\left(\sum_{i=1}^m t_i A_i\right) \text{ implies } f^T z = 0 \text{ whenever } \left(\sum_{i=1}^m t_i A_i\right) z = 0.$$

This shows that x only acts as a dummy variable in (5): each x with $(\sum_{i=1}^m t_i A_i)x = f$ yields the same value $f^T x$. Hence (1.5) reduces to

$$(1.6) \quad \min_{t \in \mathbb{R}^m} \frac{1}{2} f^T x, \quad \text{where } x \text{ solves } \left(\sum_{i=1}^m t_i A_i\right) x = f$$

$$\text{for } t \text{ with } t_i \geq 0, \quad i = 1, \dots, m \quad \text{and} \quad \sum_{i=1}^m t_i = 1.$$

A similar arguing yields for the inner minimization in (1.4) and for each fixed t with $t_i \geq 0$, $i = 1, \dots, m$,

$$\min_{z \in \mathbb{R}^n} \Pi(t, z) = \begin{cases} -\infty & \text{if } f \notin \text{range}\left(\sum_{i=1}^m t_i A_i\right), \\ -\frac{1}{2} f^T x, & \text{where } x \text{ solves } \left(\sum_{i=1}^m t_i A_i\right) x = f, \text{ otherwise.} \end{cases}$$

The equivalence of (1.4) and (1.5) is an immediate consequence. The philosophy of the ground-structure approach is to mimic, by a huge number of nodes in the starting layout, “moves” of the nodal positions also. Consider, e.g., refinement of the example from Figure 1.1. In each coordinate we multiply the number of nodes by three. This results in 403 nodes and 49414 potential bars. Figure 1.2 shows the initial layout (where only some of the bars originating from the south-west node are depicted) and the optimal truss. In real world applications, $n = 1000$ and $m = 100,000$ are typical numbers of nodes and bars in our context. It is well-known that standard optimization software fails to find an optimum of (1.5), even for moderate n and m . The crucial step to solve (1.5) (or, equivalently, (1.4)) lies in an appropriate reformulation. Using a classic minimax-theorem, we can switch the order of “max” and “min” in (1.4). Some further simple analysis gives

$$\min_{x \in \mathbb{R}^n} \max_{\substack{t \in \mathbb{R}^m, t_i \geq 0 \\ \sum_{i=1}^m t_i = 1}} \frac{1}{2} \left((b_1^T x)^2, \dots, (b_m^T x)^2 \right) \begin{pmatrix} t_1 \\ \vdots \\ t_m \end{pmatrix} - f^T x,$$

and thus by standard linear programming (LP)-theory

$$(1.7) \quad \min_{x \in \mathbb{R}^n} \max_{1 \leq i \leq m} \left\{ \frac{1}{2} (b_i^T x)^2 - f^T x \right\}.$$

Hence we have arrived at a convex problem in x with dimension n instead of $m + n$, where m was of order n^2 ; cf. Ben-Tal and Bendsøe [3] or Kočvara and Zowe [14] for details.

Problem (1.7) is a nonsmooth problem to which one can apply nonsmooth software; this has been done, e.g., in [1]. However, it proved to be more profitable to work with a reformulation of (1.7) as a constrained but smooth problem:

$$(1.8) \quad \min_{x \in \mathbb{R}^n, \alpha \in \mathbb{R}} -\alpha - f^T x \quad \text{subject to} \quad \frac{1}{2} (b_i^T x)^2 \leq -\alpha, \quad i = 1, 2, \dots, m.$$

It is shown in [14, 3] that the optimal x^* from (1.8) together with a corresponding Lagrange vector of the constraints in (1.8) is always an optimal solution of (1.4) and (1.5). Ben-Tal and Roth [4] and Ben-Tal and Zibulevsky [5] proposed to apply modern log barrier and modified barrier methods to (1.8). Our purpose is to adapt a state-of-the-art interior-point method for linear programs to solve (1.8) (and thus (1.4) and (1.5)). We add that both these approaches provide a Lagrange multiplier (and thus an optimal design vector t^* !) “for free” when solving (1.8).

Let us note that (1.8) can even be reduced to a linear program (see Achtziger et al. [1]). On purpose we do not make use of this very special LP-feature which is lost when we switch to more general problems, like the multiload case or the optimization of material. In the multiload case, one has slightly more general constraints:

$$\frac{1}{2} \sum_{j=1}^k (b_{ij}^T x)^2 \leq -\alpha, \quad i = 1, 2, \dots, m,$$

where k is the number of load vectors. In this case, an LP-reduction is no longer possible. The same is true for material optimization where the b_i 's are no longer vectors but rectangular matrices. To include also these cases in our numerical approach, we stick to the formulation (1.8) which is easily generalizable to these situations.

1.2. Notation. The following notation will be used from now on. Vectors are denoted by letters like x, s, t and by $b_1, \dots, b_m$. If the letter s denotes a vector, the diagonal matrix with diagonal s will be denoted by the capital letter S , and the diagonal matrix with diagonal t will be denoted by T . By $\|x\|$ we denote the Euclidean norm of a vector x . The iteration index k is denoted by a superscript for vectors like x^k and by a subscript for scalar quantities like α_k . Subscripts for vectors denote certain components of the vector, e.g., s_i denotes the i th component of s . Further, we write $t \geq 0$ ($t > 0$) for a vector t if $t_i \geq 0$ ($t_i > 0$) for all its components. By e we denote the vector of all ones; its dimension will always be obvious from the context.

1.3. Outline. In the next section we present an interior-point method for solving convex programs. In section 3, we discuss some special features of problem (1.8) when applying the interior-point method from section 2. In section 4 we establish global convergence of the long step method from section 2, and in section 5 we present some numerical results.

2. A primal-dual interior-point method. Polynomial time versions of interior-point methods for solving quadratically constrained convex programs like (1.8) were proposed in 1987 by Jarre [11] and independently by Mehrotra and Sun [21] in 1988. Subsequently, much more general classes of convex problems have been considered in the context of polynomial time interior-point methods in the recent past, the most general approach being that of Nesterov and Nemirovskii [23]. The papers [11, 21, 23], however, are mostly of theoretical interest, and the development of practical methods for problems like (1.8) differs from these papers.

Since the theoretical analysis of interior-point methods for problem (1.8) suggests that (1.8) is of the same complexity as a linear program (both need about the same number of iterations to reduce the “duality gap” by a factor of 2; see, e.g., [11]), one might be tempted to apply the same algorithm (suitably adapted to handle quadratic constraints) to both problems. This was done, for example, in [12], and the observation based on some preliminary numerical results was that for a primal path-following predictor-corrector method, the number of iterations for solving quadratically constrained problems was even slightly lower than the number of iterations for solving linear programs.

On the other hand, the method of choice for solving linear programs is a primal-dual version of the interior-point method, and this leads us to consider primal-dual interior-point methods for problem (1.8) as well. Our goal is to find a practical primal-dual method for the following convex program:

$$(2.1) \quad \text{minimize } c^T x \quad \text{subject to } g_i(x) \leq 0 \quad \text{for } 1 \leq i \leq m,$$

where $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ are C^3 -smooth convex functions. Note that (1.8) is a special case of (2.1).

We use a simple modification of Newton’s method for solving the KKT conditions of (2.1), much in the spirit of interior-point methods for linear programs developed, for example, in Kojima, Mizuno, and Yoshise [16], Megiddo [19], Monteiro and Adler [22], and Tanabe [26]. Our focus will be on the main differences of interior-point methods for convex problems and methods for solving linear programs as presented in the above papers. With some further modifications such methods are also generalizable to arbitrary smooth nonconvex programming problems; see, for example, the recent paper [7].

Observe that the KKT conditions for (2.1) are given by

$$(2.2) \quad \begin{cases} c + \sum_{i=1}^m t_i \nabla g_i(x) = 0, \\ g_i(x) + s_i = 0 \quad (1 \leq i \leq m), \\ s_i t_i = 0 \quad (1 \leq i \leq m), \\ s_i, t_i \geq 0. \end{cases}$$

As in most interior-point methods, the complementarity conditions $s_i t_i = 0$ are relaxed to

$$(2.3) \quad s_i t_i = r \quad \text{for } 1 \leq i \leq m$$

with some small positive parameter r , and a damped version of Newton’s method is applied to solve (2.2) iteratively, while maintaining the strict inequalities $s_i^k > 0$ and

$t_i^k > 0$ for the iterates s^k, t^k . Linearization of the relaxed KKT conditions results in a linear system

$$(2.4) \quad \begin{pmatrix} \sum t_i \nabla^2 g_i(x) & \nabla g(x) \\ Dg(x) & I \\ T & S \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta s \\ \Delta t \end{pmatrix} = \begin{pmatrix} -c - \sum t_i \nabla g_i(x) \\ (-g_i(x) - s_i)_i \\ re - St \end{pmatrix},$$

where S and T are diagonal matrices (the diagonals given by the vectors s and t , as explained in section 1.2), $g(x)$ is the vector with components $g_i(x)$, and $\nabla g(x)$ is the transpose of the Jacobian $Dg(x)$. Given a fixed starting point x^0 , we define the following weighted residuals at the point (x, s, t) with $s, t > 0$:

$$(2.5) \quad \text{res}_1(x, t) = \frac{1}{\|c\|} \left(c + \sum_{i=1}^m t_i \nabla g_i(x) \right),$$

$$(2.6) \quad \text{res}_2(x, s) = \frac{1}{1 + \|g(x^0)\|} (g(x) + s),$$

$$(2.7) \quad \text{res}_3(s, t) = \frac{St}{\|c\| (1 + \|g(x^0)\|)}.$$

To measure the distance from optimality of some iterate (x, s, t) , we define a function $\mathbf{d}$ in terms of the above residuals as

$$\mathbf{d}(x, s, t) = \|\text{res}_1(x, t)\| + \|\text{res}_2(x, s)\| + \|\text{res}_3(s, t)\|_1.$$

We now outline a primal-dual interior-point algorithm.

ALGORITHM 2.1 (primal-dual interior-point algorithm).

INPUT: An initial point x^0 , $s^0 > 0$, $t^0 > 0$, and a stopping tolerance $\epsilon > 0$.

For $k = 0, 1, \dots$, do :

1) If $\mathbf{d}(x^k, s^k, t^k) \leq \epsilon$, then stop.

2) Choose $\sigma_k \in (0, 1)$.

3) Compute $\Delta x, \Delta s$, and Δt from the linear system (2.4) with

$$(2.8) \quad x = x^k, \quad s = s^k, \quad t = t^k, \quad \text{and} \quad r = \sigma_k \frac{s^T t}{m}.$$

4) Compute

$$(2.9) \quad \bar{\lambda}_t^k := \max\{\lambda \leq 1 \mid t^k + \lambda \Delta t \geq 0\} \text{ and } \bar{\lambda}_s^k := \max\{\lambda \leq 1 \mid s^k + \lambda \Delta s \geq 0\}.$$

5) Choose $\rho_k \in (0, 1)$.

6) Set

$$\lambda_t^k = \rho_k \bar{\lambda}_t^k \quad \text{and} \quad \lambda_s^k = \rho_k \bar{\lambda}_s^k \quad \text{and} \quad \lambda_x^k = (\lambda_t^k + \lambda_s^k)/2.$$

7) Set

$$x^{k+1} = x^k + \lambda_x^k \Delta x,$$

$$s^{k+1} = s^k + \lambda_s^k \Delta s,$$

$$t^{k+1} = t^k + \lambda_t^k \Delta t.$$

TABLE 2.1

	$\rho \geq 0.9$ Default start	$\rho \geq 0.9$ Small s^0	$\rho \geq 0.5$ Small s^0
Corrector and diff. steps	10	19	20
Corrector and same steps	12	60	82
No corrector and diff. steps	18	$\gg 100$	49
No corrector and same steps	22	$\gg 100$	92

2.1. Differences to linear programs. The only degrees of freedom that are left in Algorithm 2.1 (apart from the choice of the starting point) are the choice of the parameter σ_k at each iteration and of the step length ρ_k that is taken along the Newton direction at each Newton step.

The choice of σ_k and ρ_k is very important for the efficiency of the above method (for the effect of ρ_k see Table 2.1). In [17], an implementation of an interior-point method for solving linear programs is discussed in detail. At each iteration, ρ_k is set to 0.9995; i.e., a step of length 99.95% to the boundary along the primal-dual Newton direction is taken. This makes sense since the residual decreases linearly and thus, the longer the step the more the infeasibility is reduced. It turns out, however, that a straightforward application of this step length strategy is not efficient (or not even convergent) for problems (1.8) or (2.1).

Why do primal-dual methods behave so differently for linear and for quadratically constrained programs while primal methods are reported to behave quite similarly for both problems (see, e.g., [12], [13])? To give a partial answer, we note that primal methods rely on the self-concordance (see [23]) of the barrier function, and self-concordance equally holds for linear and convex quadratic constraints. Primal-dual methods further exploit the structure of the problem, implicitly using the self-scaling property recently identified by Nesterov and Todd [24]. While both methods are of the same worst case complexity, primal-dual methods are reported to converge much faster for linear programs [25]. Two of the key points used in the implementation of primal-dual methods for solving linear programs are the following.

- The use of different step lengths in the primal and the dual space due to the fact that the primal and dual residuals depend *only* on the primal and dual variables, respectively.
- The use of *very long steps* due to the fact that both primal and dual residuals are linear functions of the step length, and a step of length 1 reduces both residuals to 0.¹

Both of these key points no longer hold true for problem (2.1). The primal and dual residuals res_1 and res_2 both depend on x , and when different steps in s and t are taken, it is not clear what step length to choose for x . (Our choice of the arithmetic mean above has no theoretical foundation.) Furthermore, both residuals are nonlinear. In particular, in our implementation a step of length 1 was possible at certain iterations, but if a step of length 1 was blindly taken, then the norm of the residuals could increase by a factor of over 100 (due to the nonlinearity) instead of going to 0 as in the case of a linear program. Sometimes it may even be necessary to increase the norm of the residuals along the search direction. If both residuals, res_1 and res_2 , happen to be very small, for example, but the parameter $r = r_k$ is large,

¹In analogy to the commonly used terminology in interior-point methods, by “very long” we denote a step of length ≤ 1 that brings the iterate close to the boundary of the feasible orthant $s \geq 0, t \geq 0$. (A step of length 1 typically violates the nonnegativity requirements $s \geq 0$ or $t \geq 0$.)

then only very short steps can be taken when reducing r to 0 if one does not allow for an increase of the residuals. (Loosely speaking, this is the case since for large r the current iterate is still “far away” from an optimal solution, and when taking a long Newton step toward the optimal solution, the residuals will necessarily increase because of their nonlinearity.) For problem (1.8), a much more careful choice of the step length is therefore necessary than it is for linear programs. Similarly, for linear programs it is reported that a rather small choice of σ in Algorithm 2.1 yields an efficient method, while our observation for problem (1.8) was that for most parts of the algorithm, large values $\sigma \geq 0.5$ were more efficient than smaller values. Only in the final iterations our algorithm “chose” small values of $\sigma \leq 0.01$ and showed a fast rate of convergence (in all examples that we tested).

2.2. Adaptation of Mehrotra’s predictor-corrector method. The most surprising experience, when testing different versions of our primal-dual method, was the observation that inclusion of a corrector step similar to the one used by Mehrotra [20] for linear programs along with different step lengths in x , s , and t greatly reduced the number of iterations. Mehrotra’s predictor step is based on a fixed point iteration derived from (2.2)–(2.3). Relations (2.2)–(2.3) can be written as

$$\Psi(z) = 0,$$

where $z = (x, s, t)$ and the function Ψ computes the left-hand sides of the equations in (2.2)–(2.3). Mehrotra’s approach can be motivated as follows: first, Newton’s method computes a direction (prediction) Δz^p from the relation

$$(2.10) \quad 0 \stackrel{!}{=} \Psi(z + \Delta z^p) = \Psi(z) + D\Psi(z)\Delta z^p + R_z(\Delta z^p)$$

by ignoring the remainder term R_z . Hence Newton direction satisfies the linear approximation to (2.10)

$$0 = \Psi(z) + D\Psi(z)\Delta z^p.$$

(This is exactly (2.4).) We note that $R_z(\Delta z^p)$ is easily computable for a given function Ψ and a given Δz^p by

$$R_z(\Delta z^p) = \Psi(z + \Delta z^p) - \Psi(z) - D\Psi(z)\Delta z^p.$$

Moreover, in our particular example (1.8), Ψ is quadratic, so that

$$(2.11) \quad R_z(\Delta z^p) = \frac{1}{2} D^2\Psi(z)[\Delta z^p, \Delta z^p].$$

Mehrotra’s corrector defines Δz as

$$(2.12) \quad \Delta z = -D\Psi(z)^{-1}(\Psi(z) + R_z(\Delta z^p)).$$

Therefore, Δz satisfies the relation

$$0 = \Psi(z) + D\Psi(z)\Delta z + R_z(\Delta z^p),$$

and in particular, if Δz^p happened to be (almost) equal to Δz , then the relation (2.10) was satisfied (almost) exactly.

We call the solution of (2.12) *Mehrotra's corrector*—even though (2.12) differs slightly from the scheme used by Mehrotra [20]—and we use this corrector by replacing the Newton direction $\Delta x, \Delta s, \Delta t$ of (2.4), in Algorithm 2.1 with the direction $\Delta x, \Delta s, \Delta t$ from (2.12). Variants of Algorithm 2.1 based on (2.4) (but using different step lengths) have been analyzed in a general framework in [23], and more recently in [27]. At this point we are not able to prove convergence for Algorithm 2.1 when the search direction is obtained by Mehrotra's corrector, nor can we prove convergence when different steps in the variables s and t are taken at each iteration. In our program we therefore included a safeguard. When the corrected Newton step does not result in a certain decrease of the residuals in (2.4), the program temporarily reverts to the plain primal-dual method with equal step sizes (and such that the step length minimizes a certain measure for the residual corresponding to $s_i t_i = r$). In our experiments the corrector step proved to be very efficient, and the safeguard of reverting to the plain primal-dual method was necessary only when the starting point was deliberately poorly chosen (very small s^0 in Table 2.1). Table 2.1 illustrates the importance of the corrector step in different versions of the algorithm. It refers to a truss topology design problem with $n = 144$, and $m = 2040$. We list the number of iterations for the method with and without corrector step as well as with and without the use of different step lengths in s and t . The first column refers to the default starting point of section 2.5, and the second column to the default starting point but with s^0 divided by 1000. In both columns, we force a step of at least 90% to the boundary of the positive orthant at each iteration, that is, $\rho_k \geq 0.9$ in the description of Algorithm 2.1. The third column refers to the same (poor) starting point as column 2 but with shorter steps ρ_k along the Newton direction.

Concerning the starting point, when one of the variables is poorly chosen (in columns 2 and 3, s^0 is closer by a factor of 1000 to the “boundary $s = 0$ ” than t^0), then the table indicates that different steps in s and t are meaningful. Column 2 of the table also shows the intuitively obvious fact that the method does not seem to be globally convergent when long steps towards the boundary are taken at each iteration regardless of the behavior of the norms of the residuals.

A qualitative analysis of Mehrotra's corrector step reveals that a slight damping of his corrector might be useful, and this observation was indeed confirmed by our experiments. In Algorithm 2.1, due to the positivity requirement, only a step of length $\rho < 1$ along Δz is taken at each iteration. For a step of length less than 1, however, it may be more efficient to define Δz as

$$\Delta z = -D\Psi(z)^{-1}(\Psi(z) + \lambda R_z(\Delta z^\rho))$$

for some value of $\lambda \in [0, 1]$. The motivation for considering values of $\lambda < 1$ arose as in one of our test runs we observed that for $\lambda = 1$ the reduction of the step length ρ from 0.99 to 0.5 significantly increased the dual residual res_2 . In this particular situation, the iterate was in the domain of superlinear convergence² of the interior-point method, and the scaled norm of the dual residual res_2 was in the order of 10^{-9} for $\rho = 0$ (i.e., the current iterate), as well as for $\rho = 1$ (the full step), but it was in the order of 10^{-5} for $\rho = 0.5$. Due to the nonlinearity, Δx happened to be a good approximation for the full step, but not for a step of length 0.5. Note that this could not happen if res_2 was linear!

²We do not prove superlinear convergence of our algorithm, but our numerical experiments unanimously reconfirm it.

We examine which values of λ may be most efficient when a step of length ρ is taken. A step with parameters λ and ρ results in the following function value for Ψ :

$$\begin{aligned} & \Psi(z - \rho D\Psi(z)^{-1}(\Psi(z) + \lambda R_z(\Delta z^p))) \\ &= (1 - \rho)\Psi(z) - \lambda\rho R_z(\Delta z^p) + R_z(\rho D\Psi(z)^{-1}(\Psi(z) + \lambda R_z(\Delta z^p))) \end{aligned}$$

(by using (2.10)). Applying (2.11) and the definition of Δz^p , this can further be reduced to

$$(2.13) \quad \begin{aligned} &= (1 - \rho)\Psi(z) + (\rho^2 - \lambda\rho)R_z(\Delta z^p) + \lambda\rho^2 D^2\Psi(z)[\Delta z^p, D\Psi(z)^{-1}R_z(\Delta z^p)] \\ &\quad + \rho^2 \lambda^2 R_z(D\Psi(z)^{-1}R_z(\Delta z^p)). \end{aligned}$$

For $\lambda = \rho = 1$, this becomes

$$D^2\Psi(z)[\Delta z^p, D\Psi(z)^{-1}R_z(\Delta z^p)] + R_z(D\Psi(z)^{-1}R_z(\Delta z^p)),$$

which is a cubic term in Δz^p . Based on this relation, it is straightforward to show that Mehrotra's corrector step is locally cubically convergent for fixed $r_k > 0$.

When taking a step of length $\rho < 1$, we cannot expect to reduce Ψ to less than $(1 - \rho)\Psi(z)$. The remaining terms in (2.13) are cubic in Δz^p if and only if λ is chosen as $\lambda = \rho$. In this case the remaining terms reduce to

$$\lambda^3 D^2\Psi(z)[\Delta z^p, D\Psi(z)^{-1}R_z(\Delta z^p)] + \lambda^4 R_z(D\Psi(z)^{-1}R_z(\Delta z^p)).$$

We point out, however, that the value of the step length ρ in our algorithm depends on the prior choice of λ (determining the search direction), and in our experiments, larger values of λ resulted in larger values of ρ and thus in faster convergence. In some examples, setting $\lambda = 0.5(1 + \rho^p)$, where ρ^p is determined from the predictor step Δx^p , improved the number of iterations by one. Overall, however, the value $\lambda = 1$ generated steps that allowed larger values of ρ without violating the positivity requirements $s > 0, t > 0$, and therefore, the optimal choice of λ might be close or equal to one.

Another option, recently proposed by Andersen and Ye [2], is to apply the Mehrotra corrector only to the complementarity condition, but not to the nonlinear equalities. This approach somewhat simulates the strategy of SQP methods that solve the complementarity condition exactly for each subproblem. In the future we plan to use a hybrid approach and to use the full Mehrotra corrector for the complementarity condition and a damped corrector term for the nonlinear equalities.

2.3. The duality parameter. As mentioned, apart from the choice of the step length, the only degree of freedom in Algorithm 2.1 is the choice of σ at each iteration. We use an adaptive strategy that depends on a number of parameters specified in some input parameter file. To simplify the presentation, we denote some of the constants by their default values rather than giving them an extra name. The value of σ at iteration k is defined as

$$(2.14) \quad \sigma_k = \begin{cases} \frac{3}{4} & \text{if } \|\text{res}_3\|_1 < (\|\text{res}_1\| + \|\text{res}_2\|)/1000, \\ \frac{1}{2} & \text{if } \|\text{res}_3\|_1 \geq (\|\text{res}_1\| + \|\text{res}_2\|)/1000 \text{ and } \beta > 30, \\ \min\{0.6, (1 - \tilde{\rho}_{k-1})_1^{\text{const}}\} & \text{else.} \end{cases}$$

In the above expression, the number $\tilde{\rho}_{k-1}$ is the actual step length taken along Δx in the previous iteration $k-1$, the number $\beta = \beta(s, t)$,

$$(2.15) \quad \beta(s, t) = \frac{s^T t}{m \min_i s_i t_i} - 1,$$

measures the size of the residual (2.3), and the quantities $\text{res}_1, \text{res}_2, \text{res}_3$ are evaluated at x^k, s^k, t^k . The constant const_1 was set to 3 in our experiments. The remaining constants used in (2.14), i.e., the numbers $\frac{3}{4}, 1000, \frac{1}{2}, 30$, and 0.6, are arbitrary in

$$(\text{const}_2, 1) \times [1, \infty) \times (0, \text{const}_2] \times (1, \infty) \times (0, \text{const}_2]$$

for some $\text{const}_2 \in (0, 1)$ and can also be changed with a parameter input file; the given numbers are merely the default values. While Mehrotra's corrector was quite insensitive to the reduction σ of r , a stronger reduction of r like $\sigma \leq 0.1$ increased the number of iterations for the plain primal-dual method. For the example in Table 2.1 of section 2.1, the method with $\sigma \leq 0.1$ took 28 iterations compared to 18 with $\sigma \leq 0.6$. The strategy of determining σ_k from $\tilde{\rho}_{k-1}$ is motivated by the observation that the convergence behavior of Algorithm 2.1 was fairly monotone, in the sense that when a long step was possible at iteration $k-1$, then, typically, the same was true for iteration k . Since long steps are taken only when σ is small, the above strategy is an adaptive rule based on the previous iteration. In the final iterations this strategy always resulted in a fast (superlinear) convergence in all our experiments.

2.4. The line search. The most delicate detail of the implementation of Algorithm 2.1 turned out to be the choice of an appropriate step length ρ at each iteration. The line search forms the longest subroutine in our program, and this fact is also reflected by the somewhat lengthy description below, where we summarize the principles used during the line search.

At first, we compute some auxiliary quantities.

- Determine $\bar{\lambda}_s$ and $\bar{\lambda}_t$ as the largest numbers that are less or equal to 1 and such that $s + \bar{\lambda}_s \Delta s \geq 0$ and $t + \bar{\lambda}_t \Delta t \geq 0$.
- Choose a constant $\text{const}_3 \in [0, 1]$ and multiply Δs by $(\bar{\lambda}_s)^{\text{const}_3}$, Δt by $(\bar{\lambda}_t)^{\text{const}_3}$, and Δx by $((\bar{\lambda}_s)^{\text{const}_3} + (\bar{\lambda}_t)^{\text{const}_3})/2$.

Define

$$\hat{\lambda}_s = (\bar{\lambda}_s)^{1-\text{const}_3}, \quad \hat{\lambda}_t = (\bar{\lambda}_t)^{1-\text{const}_3}.$$

Here, $\hat{\lambda}_s$ and $\hat{\lambda}_t$ are the maximum feasible steps along (the new) Δs and Δt . We use this rescaling of the search direction to be able to take different steps in the variables s and t (namely, when $\text{const}_3 > 0$), while formally still taking a step of the same length in all variables. Only in the case that $\beta > 30$ (see (2.15)), we restrict ourselves to equal steps in x and s ($\text{const}_3 = 0$). In our test runs, the choice $\text{const}_3 = 1$ overall resulted in the fastest algorithm.

- Next, investigate the step $(x^{k+1}, s^{k+1}, t^{k+1}) = (x^k, s^k, t^k) + \rho(\Delta x^k, \Delta s^k, \Delta t^k)$, and find the smallest step length $\rho = \hat{\rho} \in (0, 1]$ that is a local minimizer of

$$(2.16) \quad \delta(\rho) := \sqrt{\text{res}_1(x + \rho \Delta x, t + \rho \Delta t)^2 + \text{res}_2(x + \rho \Delta x, s + \rho \Delta s)^2}.$$

Define the reduction of the residuals resulting from a step of length $\hat{\rho}$ by

$$(2.17) \quad \bar{\sigma} = \delta(\hat{\rho})/\delta(0).$$

By our choice of the Euclidean norm, finding $\hat{\rho}$ is cheap. If all constraints are quadratic, it amounts to minimizing just one scalar polynomial of degree 4. Above, the quantity $\hat{\rho}$ refers to the same step length in Δs and Δt . Note, however, that Δs and Δt may have been rescaled if $\text{const}_3 > 0$.

The step length $\tilde{\rho}$ used for the update

$$(x^{k+1}, s^{k+1}, t^{k+1}) = (x^k, s^k, t^k) + \tilde{\rho}(\Delta x^k, \Delta s^k, \Delta t^k)$$

is defined as the maximum step that satisfies the following restrictions.

1. Allow long steps only in the case when the primal and dual residuals are reduced significantly. More precisely, restrict $\tilde{\rho} \leq \hat{\rho}$ (see (2.16)) and $\tilde{\rho} \leq \bar{\sigma} \min\{\hat{\lambda}_t, \hat{\lambda}_s\}$ unless the norm of $\text{res}_3(s, t)$ happens to be larger than the sum of the norms of $\text{res}_1(x, t)$ and $\text{res}_2(x, s)$. In the latter case allow for a moderate³ increase of the norms of $\text{res}_1(x, t)$ and $\text{res}_2(x, s)$.
2. Restrict $\tilde{\rho} \leq (1 - \sigma) \min\{\hat{\lambda}_t, \hat{\lambda}_s\}$; i.e., allow long steps only in the case when the duality parameter r has been significantly reduced.
3. Restrict $\tilde{\rho} \leq \bar{\rho}$, where $\bar{\rho}$ is the largest number in $[0, \min\{\hat{\lambda}_s, \hat{\lambda}_t\}]$ such that $\beta(s + \rho\Delta s, t + \rho\Delta t) \leq 60$ for all $\rho \in [0, \bar{\rho}]$. Here, 60 is a default value larger than the value 30 in the definition (2.14) of σ_k , and β is as in (2.15).
4. If $\|\text{res}_3\|_1 < (\|\text{res}_1\| + \|\text{res}_2\|)/1000$, use the following safeguard: set $\text{const}_3 = 0$ and restrict $\tilde{\rho} \leq \check{\rho}$, where $\check{\rho}$ is the largest number in $[0, 1]$ with

$$\frac{(s + \rho\Delta s)^T(t + \rho\Delta t)}{s^T t} \geq \frac{\delta(\rho)}{\delta(0)} \quad \text{for all } \rho \in [0, \check{\rho}].$$

The number 1000 is the same default value as in (2.14).

2.5. The starting point. We conclude this section by outlining our procedure for finding a starting point. We focus our attention to the special case of problem (1.8) and point out some extensions to the general case of (2.1). First, fix x^0 (e.g., $x^0 = 0$) and set

$$H = \sum_{i=1}^m \nabla^2 g_i(x^0) + \nabla g_i(x^0) \nabla g_i(x^0)^T \quad \text{and} \quad \nu = \sqrt{c^T H^{-1} c}.$$

In the notation of (1.8) this is

$$H = \sum_{i=1}^m b_i b_i^T \quad \text{and} \quad \nu = \sqrt{f^T H^{-1} f}.$$

Note that H is nonsingular because of (1.2) and thus ν is well defined. Then define $\hat{r} = \sqrt{m}\nu$, $t^0 = 3e/n$, and $r_0 = \|t^0\|_\infty \hat{r}$. Finally, set $s^0 = \hat{r}e$. Obviously, $(t^0)^T s^0 / m = r_0$.

In the above procedure, the factorization of H used to compute ν can be used in the first iteration of the algorithm so that the cost of finding the starting point is essentially $O(m)$.

We note that the seemingly better choice of $t^0 = e/m$, satisfying relation (3.1) and preserving $r_0 = \hat{r}$, turned out to be slightly less efficient in all tested examples.

³We suppress the details of how to interpret “moderate”; we assume, however, that the norms of res_1 and res_2 remain bounded.

Our choice above was motivated by the idea that, at the optimal solution, the largest components of t are at least of the size $1/n$, and that it is “better” to start with a too large starting point rather than with a too small one. The constant 3 in the above definition of t is the default value used in our numerical experiments and can be changed by an input parameter file.

The choice of the initial duality parameter $\hat{r}$ is motivated by the primal problem. For problem (1.8) the initial point $x^0 = 0$ is the analytic center of the (primal) feasible set. Thus, if $\hat{r}$ was set to $\hat{r} = m\nu$, the H^{-1} -norm of the initial gradient of the primal problem was equal to one. If the H^{-1} -norm was less than $1/4$, for example, the theory of self-concordant functions would imply quadratic convergence of Newton’s method to the analytic center. However, since the theory is overly pessimistic in general, we reduce $\hat{r}$ by a factor of $\sqrt{m}$. Again, the factor of $\sqrt{m}$ is an input parameter and can be changed to any power of m . For a general problem (2.1) for which $x^0 = 0$ is not a central point or not even feasible, it seems meaningful to add a term to $\hat{r}$ that models the infeasibility and to define, for example,

$$\hat{r} = \sqrt{m}\nu + \max\{0, \max_i \{g_i(x^0) + \sqrt{m}\nu\}\}.$$

(For (1.8) this is the same value as before.) The initial value of t for a general problem (2.1) can be chosen, for example, as a multiple of the all-ones-vector that minimizes the square norm of $\text{res}_1(x^0, s^0, t^0)$ plus a multiple of the barrier term $-\sum \ln t_i$. (If the multiple is chosen as $18/n^2 - 6/(nm)$ we obtain the starting vector $t^0 = 3e/n$ for problem (1.8) above.)

The numerical example in section 2.2 referred to the above starting point and to the same starting point with r_0 and s^0 divided by 1000.

3. The special structure of problem (1.8). The KKT conditions for problem (1.8) can be written as follows: there exist vectors $s, t \in \mathbb{R}^m$, $s, t \geq 0$ such that

$$(3.1) \quad \sum_{i=1}^m t_i = 1,$$

$$(3.2) \quad \sum_{i=1}^m t_i (b_i^T x) b_i = f,$$

$$(3.3) \quad \frac{1}{2} (b_i^T x)^2 + \alpha + s_i = 0 \quad \text{for } 1 \leq i \leq m,$$

$$(3.4) \quad s_i t_i = 0 \quad \text{for } 1 \leq i \leq m.$$

Note that Slater’s condition is trivially satisfied for (1.8), and hence the above conditions are necessary and sufficient for optimality.

3.1. Scaling the problem. Let us now analyze the properties that follow from the special structure of our problem and, in particular, its sensitivity to scaling of the data f , b_i . We identify linear transformations of the data f and b_i ($1 \leq i \leq m$) in (1.8), such that the transformed problem is of the same structure as the original problem, and such that the solution of the original problem can immediately be recovered from the solution of the transformed problem. Such problems are called “equivalent.” The transformations will also affect the quantities involved in the KKT conditions (3.1)–(3.4).

A natural choice to measure the closeness of some tuple $\alpha, x, s \geq 0, t \geq 0$ to an optimal solution of (1.8) might seem to require that conditions (3.1)–(3.4) hold up to

some tolerance ϵ . This, of course, is meaningful only if we can somehow eliminate the dependence of (3.1)–(3.4) on linear transformations that lead to equivalent problems. For this purpose we will introduce in the next section a “distance” function which is the balanced sum of the residuals of (3.1)–(3.4) and which is in some sense balanced under the following linear transformation leading to equivalent variants of (1.8).

In general, we may consider an arbitrary linear transformation of the variables x, α in (1.8). Transformations, however, that maintain the special structure of (1.8), namely, separability of x and α in all constraints, are of the form $\hat{x} := U^{-T}x$ and $\hat{\alpha} := \rho_1^{-1}\alpha$ with some nonsingular matrix U and some positive scalar ρ_1 . Since the solution of the problem does not change when the objective function is multiplied by a positive scalar ρ_2 and the constraints are multiplied by positive scalars d_i , we obtain the following class of problems that are equivalent to (1.8) and depend on ρ_1 , U , and on additional parameters $\rho_2 > 0$, $d_i > 0$ ($1 \leq i \leq m$),

$$(3.5) \quad \min_{\hat{x} \in \mathbb{R}^n, \hat{\alpha} \in \mathbb{R}} \left\{ -\rho_2 \rho_1 \hat{\alpha} - \rho_2 (Uf)^T \hat{x} \mid d_i \left(\frac{1}{2} ((Ub_i)^T \hat{x})^2 + \rho_1 \hat{\alpha} \right) \leq 0 \right\}.$$

For a solution $\hat{x}, \hat{\alpha}$ of (3.5), a solution to (1.8) is given by $x = U^T \hat{x}$ and $\alpha = \rho_1 \hat{\alpha}$. The KKT conditions for (3.5) are

$$(3.6) \quad \begin{aligned} \sum_{i=1}^m \rho_1 d_i \hat{t}_i &= \rho_1 \rho_2, \\ \sum_{i=1}^m d_i \hat{t}_i ((Ub_i)^T \hat{x}) Ub_i &= \rho_2 Uf, \\ d_i \left(\frac{1}{2} ((Ub_i)^T \hat{x})^2 + \rho_1 \hat{\alpha} \right) + \hat{s}_i &= 0 \quad \text{for } 1 \leq i \leq m, \\ \hat{s}_i \hat{t}_i &= 0 \quad \text{for } 1 \leq i \leq m. \end{aligned}$$

Clearly, for a given (x, α, s, t) and its transformed counterpart

$$(\hat{x}, \hat{\alpha}, \hat{s}, \hat{t}) = (U^{-T}x, \rho_1^{-1}\alpha, Ds, \rho_2 D^{-1}t),$$

the norms of the residuals (3.1)–(3.4) may differ a lot depending on the choice of ρ_1, ρ_2, U and $D = \text{diag}(d_i)$.

3.2. Measuring closeness to optimality. Next we discuss suitable proximity measures for the “distance” from some point x, α, s , and t to the optimal solution of (1.8). A desirable property for the measure $\mathbf{d}$ is certainly that it should not change under the above linear transformations. On the other hand, we search for a simple measure based on some weighted norms of the residuals of (3.2)–(3.4). Here, the residual of (3.1) is neglected since (3.1) is the only linear equation among (3.1)–(3.4) and its residual is reduced the fastest in our interior-point method below. (Note that, given an approximate solution (x, α, s, t) of (3.1)–(3.4), by setting $\rho_2 = \sum t_i$, $\rho_1 = 1$, $U = I$, and $D = I$, we obtain an equivalent problem (3.5) for which $(\rho_2 x, \rho_2^2 \alpha, \rho_2^2 s, t)$ is an approximate solution and the residual of (3.1) (resp., (3.6)) is zero while the residuals of (3.2)–(3.4) are multiplied by ρ_2 (resp., ρ_2^2).) Since Algorithm 2.1 generates only iterates that satisfy $s, t > 0$, we also restrict our measure to points with

$$s, t > 0,$$

and try to identify suitable weights and norms for the residuals of (3.2)–(3.4).

As we will use the measure in a line search, norms induced by some simple scalar product are most suitable. We assume that $\|f\| \neq 0$, and measure (3.2) by $\frac{1}{\|f\|}$ times the Euclidean norm which effects that the right-hand side of (3.2) has size 1. We would like to choose the remaining norms such that the weights of (3.2), (3.3), and (3.4) are “balanced” in some sense. Rather than normalizing the residual of (3.3) as in (2.6), we use the Euclidean norm of (3.3) divided by $\|s\|$ (note that $s > 0$, and hence $\|s\| \neq 0$). Since s_i and t_i are assumed to be positive and since $s^T t$ measures the duality gap (if the other residuals are zero), a suitably scaled multiple of $s^T t$ seems appropriate as a measure for (3.4). Choosing $\frac{s^T t}{\|s\|\|t\|}$, i.e., the cosine of the angle between s and t , results in a measure that is at most 1 and is thus⁴ consistent with the measures for (3.2) and (3.3). For problem (1.8) we use the normalization of res_2 by $\frac{1}{\|s\|}$ in (3.7) instead of the one in (2.6) since it does not depend on the (arbitrary) choice of a starting point. For a general convergence analysis, however, the definition (2.6) is more appropriate. Likewise for the normalization of res_3 . By further neglecting α due to its special role, the resulting distance function used in our stopping test is

$$(3.7) \quad \mathbf{d} = \mathbf{d}(\alpha, x, s, t) := \frac{1}{\|f\|} \left\| f - \sum_{i=1}^m t_i (b_i^T x) b_i \right\| + \frac{1}{\|s\|} \left\| \begin{bmatrix} \vdots \\ s_i + \frac{1}{2} (b_i^T x)^2 + \alpha \\ \vdots \end{bmatrix} \right\| + \frac{s^T t}{\|s\| \|t\|}.$$

It is easy to verify that $\mathbf{d}$ is invariant under the above linear transformations if U in (3.5) is unitary and D is a positive multiple of the identity matrix. (The more ambitious goal to find a measure, that is invariant for all invertible U and diagonal $D > 0$ is not satisfied by the above simple measure $\mathbf{d}$.)

3.3. The linear systems for problem (1.8). To simplify the description of Newton’s method, we use the notation

$$\tilde{x} = \begin{bmatrix} \alpha \\ x \end{bmatrix}, \quad \Delta \tilde{x} = \begin{bmatrix} \Delta \alpha \\ \Delta x \end{bmatrix}$$

from now on. Given some iterate $\tilde{x}$, $s > 0$, $t > 0$ and some “centering parameter” $r = r_k$, it is straightforward to derive that the Newton direction for (3.1)–(3.3), (2.3) is given by the solution $\Delta \tilde{x}, \Delta t, \Delta s$ of

$$(3.8) \quad \begin{array}{rclcl} Q \Delta \tilde{x} & + & A \Delta t & & = \text{Res}^{(1)}, \\ A^T \Delta \tilde{x} & & & + \Delta s & = \text{Res}^{(2)}, \\ & & S \Delta t & + & T \Delta s = \text{Res}^{(3)}, \end{array}$$

where S and T are diagonal matrices (the diagonals given by the vectors s and t , as explained in section 1.2), and

$$Q = Q(t) = \sum_{i=1}^m t_i \begin{bmatrix} 0 \\ b_i \end{bmatrix} \begin{bmatrix} 0 \\ b_i \end{bmatrix}^T, \quad A = A(x) = \left[\begin{bmatrix} 1 \\ (b_1^T x) b_1 \end{bmatrix} \cdots \begin{bmatrix} 1 \\ (b_m^T x) b_m \end{bmatrix} \right],$$

⁴It is somewhat disturbing that the function $\cos(s, t)$ has slope zero at the (standard) initial values $s = s^0 = \rho_1 e$, $t = t^0 = \rho_2 e$ for some $\rho_1, \rho_2 > 0$. To “regularize” this behavior for parallel s, t , one might therefore consider “ $1 - \frac{2}{\pi} \arccos \frac{s^T t}{\|s\| \|t\|}$ ” instead of $\frac{s^T t}{\|s\| \|t\|}$.

$$\text{Res}^{(1)} = \begin{bmatrix} \text{Res}^{(1,1)} \\ \text{Res}^{(1,2)} \end{bmatrix} = \begin{bmatrix} 1 - \sum_{i=1}^m t_i \\ f - \sum_{i=1}^m t_i (b_i^T x) b_i \end{bmatrix},$$

$$(3.9) \quad \text{Res}_i^{(2)} = -s_i - \alpha - \frac{1}{2} (b_i^T x)^2 \quad \text{for } 1 \leq i \leq m,$$

and

$$\text{Res}_i^{(3)} = r - s_i t_i \quad \text{for } 1 \leq i \leq m.$$

Since T and S are diagonal, it is efficient to reduce (3.8) to the smaller system

$$(3.10) \quad (Q + ATS^{-1}A^T)\Delta\tilde{x} = \text{Res}^{(1)} + AS^{-1}(T\text{Res}^{(2)} - \text{Res}^{(3)}).$$

Note that due to (1.3) this is a positive definite system. Recovering the vectors Δs and Δt from $\Delta\tilde{x}$ can be done by using the *stable reduction* to the Schur complement introduced in [9]. The systems matrix in (3.8) is very large and sparse, while the matrix in (3.10) is of moderate dimension but rather dense. We therefore used sparse data structures for A and a full matrix for $Q + ATS^{-1}A^T$ and its Cholesky decomposition.

4. Convergence analysis. A proof of global convergence of a potential reduction interior-point method for convex programs (and certain classes of constrained equations) is given, for example, in [27]. The proof below is intended to provide a guarantee that the “technical details” of this paper are consistent, in particular, to show that the line search in section 2.4 allows a proof of global convergence contrary to the numerical example in section 2.2, where the result of the line search was overruled by the demand to go at least 90% towards the boundary, a practice that is efficient for linear programs [18].

For simplicity of presentation, we restrict our analysis to the case of equal step lengths in the variables s and t ($\text{const}_3 = 0$) and assume that the safeguard rule (4.1) below is applied. The formulation of the safeguard is more complicated for the case of different steps in Algorithm 2.1 and is therefore omitted in section 2.4. Nevertheless, our proof below can be extended to this situation analogously as in (5b), (6b) of Kojima, Megiddo, and Mizuno [15].

We further restrict our analysis to the case where the following assumptions are satisfied.

- (A) For $s, t > 0$ the linear systems (2.4) are nonsingular.
- (B) The iterates generated by Algorithm 2.1 are bounded.

Assumption (B) is the weak point in our analysis; in general it will be difficult to verify this assumption. Nevertheless, as far as we know, all proofs of convergence of long step interior-point methods in the literature use Assumption (B) (and provide sufficient criteria as to when (B) is satisfied). Also in Lemma 4.1 below, we will show that both assumptions (A) and (B) are indeed satisfied for problem (1.8).

In addition to the line search requirements in section 2.4, we further require the descent condition (5b) of Kojima, Megiddo, and Mizuno [15] for the line search⁵:

- Let $\bar{\rho}$ be the max. number $\leq \min\{\bar{\rho}, \hat{\rho}, (1 - \sigma) \min\{\hat{\lambda}_t, \hat{\lambda}_s\}, \bar{\sigma} \min\{\hat{\lambda}_t, \hat{\lambda}_s\}\}$ such that

$$(4.1) \quad (s + \rho\Delta s)^T(t + \rho\Delta t) \leq \left(1 - \rho \left(1 - \frac{1 + \sigma_k}{2}\right)\right) s^T t$$

⁵The paper [15] allows for different steps in s and t (only for linear programs) and therefore complements (5b) by a second condition (6b).

for all $\rho \in [0, \tilde{\rho}]$.

LEMMA 4.1. *Assumptions (A) and (B) are satisfied for problem (1.8) if the set of b_i ($1 \leq i \leq m$) spans all of $\mathbb{R}^n$.*

Proof. Regularity of the systems defining the Newton direction follows from the formulation (3.10) and the fact that the b_i span all of $\mathbb{R}^n$.

Since condition (3.1) is linear, it follows from Newton's method that

$$(4.2) \quad \sum_{i=1}^m t_i^k \in \left[\min \left\{ 1, \sum_{i=1}^m t_i^0 \right\}, \max \left\{ 1, \sum_{i=1}^m t_i^0 \right\} \right]$$

for all $k \geq 0$ and, by nonnegativity, the multipliers t^k are bounded. For simplicity of notation, we tighten relation (4.2) and assume from now on $\sum_{i=1}^m t_i^k = 1$ for all k .

By (4.1), $(s^k)^T t^k$ is decreasing, $(s^k)^T t^k \leq M$ for all k and some sufficiently large fixed value M . Likewise, by the choice of $\tilde{\rho}$ in the line search, the Euclidean norm of $\begin{pmatrix} \text{res}_1 \\ \text{res}_2 \end{pmatrix}$ is bounded for all iterations k . By enlarging M , we may assume without loss of generality that M is also an upper bound for the norms of the residuals of (3.2) and (3.3).

If α_k is bounded below, then s^k is bounded (else res_2 in (3.3), (2.6) became unbounded) and, likewise, x^k is bounded because else $(b_i^T x)^2$ was unbounded for at least one index i and again res_2 became unbounded.

If α_k is unbounded below, we will derive a contradiction. This contradiction is somewhat tedious to develop and fills the remainder of this proof: let k_j be a subsequence such that $\alpha_{k_j} \leq -j^2$. For $j > 2M$ it follows that

$$(4.3) \quad \frac{1}{2}(b_i^T x^{k_j})^2 \leq -\left(1 + \frac{1}{j}\right) \alpha_{k_j} \quad \text{for all } i,$$

because if there existed $\bar{i}$ violating (4.3), then

$$(4.4) \quad \|\text{res}_2\| = \left\| \left(\frac{1}{2}(b_i^T x^{k_j})^2 + \alpha_{k_j} + s_i^{k_j} \right)_{i=1 \dots m} \right\| \geq \frac{1}{2}(b_{\bar{i}}^T x^{k_j})^2 + \alpha_{k_j} \geq j > 2M,$$

which is a contradiction. Tightening (4.3) to the stronger inequality

$$\frac{1}{2}(b_i^T x^{k_j})^2 \leq -\left(1 - \frac{1}{j}\right) \alpha_{k_j} \quad \text{for all } i,$$

however, is wrong. If it was true and

1. if $s_i^{k_j} \geq \frac{1}{2}j$ for all i , then from $\sum t_i = 1$ it follows that $(s^{k_j})^T t^{k_j} \geq \frac{1}{2}j > M$, which is a contradiction, and
2. if $s_{\bar{i}}^{k_j} \leq \frac{1}{2}j$ for some $\bar{i}$, then as in (4.4),

$$(4.5) \quad \left\| \left(\frac{1}{2}(b_i^T x^{k_j})^2 + \alpha_{k_j} + s_i^{k_j} \right)_{i=1 \dots m} \right\| \geq \frac{1}{2}j > M$$

is a contradiction.

Hence there exists a nonempty set $I_j \subset \{1, \dots, n\}$ with

$$(4.6) \quad \frac{1}{2}(b_i^T x^{k_j})^2 \in \left[-\left(1 - \frac{1}{j}\right) \alpha_{k_j}, -\left(1 + \frac{1}{j}\right) \alpha_{k_j} \right] \quad \text{for } i \in I_j,$$

and $\frac{1}{2}(b_i^T x^{k_j})^2 < -(1 - \frac{1}{j})\alpha_{k_j}$ for $i \notin I_j$. Since there are only finitely many different index sets I_j , there is a subsequence j_l for which $I_{j_l} \equiv \bar{I}$ is constant. By taking another subsequence, if necessary, we may assume that for fixed $i \in \bar{I}$ the sign of $b_i^T x^{k_{j_l}}$ is constant for all $j = j_l$. When changing from b_i to $-b_i$, the problem (1.8) does not change, and without loss of generality we therefore assume $b_i^T x^{k_{j_l}} \geq 0$ for all $i \in \bar{I}$ and all natural numbers l .

To simplify the notation, we omit the index j and write l in place of j_l from now on. For $i \notin \bar{I}$ follows $s_i^{k_l} \geq l/2$ by (4.5). From the bound on res_3 it follows that $t_i^{k_l} \leq 2M/l$ for $i \notin \bar{I}$. We obtain

$$\begin{aligned} M &\geq \left\| f - \sum_{i \in \bar{I}} t_i^{k_l} (b_i^T x^{k_l}) b_i - \sum_{i \notin \bar{I}} \underbrace{t_i^{k_l}}_{\leq 2M/l \leq \sqrt{2|\alpha_{k_l}|}} \underbrace{(b_i^T x^{k_l})}_{\geq 0} b_i \right\| \\ &\geq \left\| \sum_{i \in \bar{I}} \underbrace{t_i^{k_l} (b_i^T x^{k_l})}_{=: \nu_i > 0} b_i \right\| - \|f\| - m \frac{2M}{l} \sqrt{2|\alpha_{k_l}|} \max_i \|b_i\| \\ &\geq \|\vec{\nu}\| \sigma_{\min}((b_i)_{i \in \bar{I}}) - \|f\| - m \frac{2M}{l} \sqrt{2|\alpha_{k_l}|} \max_i \|b_i\|, \end{aligned}$$

where $\sigma_{\min}$ denotes the minimal singular value of a matrix, and $\vec{\nu}$ is the vector with components ν_i for $i \in \bar{I}$. Since $b_i^T x^{k_l} \geq \sqrt{|\alpha_{k_l}|}$ and

$$\max_{i \in \bar{I}} t_i^{k_l} \geq \left(\underbrace{\sum_{i=1}^m t_i^{k_l}}_{=1} - (m - |\bar{I}|) \underbrace{\frac{2M}{l}}_{\text{bound for } t_i, i \notin \bar{I}} \right) / |\bar{I}| \geq \epsilon > 0$$

for sufficiently large l , the norm $\|\vec{\nu}\| \geq \epsilon \sqrt{|\alpha_{k_l}|}$ grows faster (when $l \rightarrow \infty$) than

$$m \frac{2M}{l} \sqrt{2|\alpha_{k_l}|} \max_i \|b_i\|,$$

and we conclude that $\sigma_{\min}((b_i)_{i \in \bar{I}}) = 0$; since $\nu_i > 0$, there exists a nonnegative vector $\vec{\lambda} \neq 0$ such that $\sum_{i \in \bar{I}} \lambda_i b_i = 0$. Now,

$$\sum_{i \in \bar{I}} \lambda_i \underbrace{b_i^T x^{k_l}}_{\in [\sqrt{2(1-\frac{1}{l})|\alpha_{k_l}|}, \sqrt{2(1+\frac{1}{l})|\alpha_{k_l}|}]} = 0$$

yields $\sum \lambda_i = 0$ (when $l \rightarrow \infty$) in contradiction to $\lambda_i \geq 0$ and $\vec{\lambda} \neq 0$. This completes our proof. $\square$

Under the assumptions (A) and (B) we now show the following theorem.

THEOREM 4.2. *The sequence of iterates (x^k, s^k, t^k) generated by Algorithm 2.1 and parameter updates as in sections 2.3 and 2.4 has accumulation points. Each accumulation point is an optimal solution of (1.8).*

(We do not explicitly assume Slater's condition. We believe, however, that assumption (A) might not be satisfied if Slater's condition is violated.)

Proof. For simplicity we use the default values of the parameters from sections 2.3 and 2.4. Since (x^k, s^k, t^k) are bounded, the existence of accumulation points is evident.

Let $(\bar{x}, \bar{s}, \bar{t})$ be an accumulation point and assume by contradiction that $\mathbf{d}(\bar{x}, \bar{s}, \bar{t}) > 0$. By considering a subsequence if necessary, we assume that the sequence of iterates $(x^k, s^k, t^k) \rightarrow (\bar{x}, \bar{s}, \bar{t})$ converges. By the safeguard in Step 4 of the line search it follows from $\mathbf{d}(\bar{x}, \bar{s}, \bar{t}) > 0$ that $\bar{s}^T \bar{t} > 0$. Let $\bar{s}^T \bar{t} = \epsilon_1$. Further, from the bound on β (2.15) guaranteed by Step 3 of the line search it follows that

$$\frac{(s^k)^T t^k}{m \min s_i^k t_i^k} - 1 \leq 60$$

for all k , and therefore $\min \bar{s}_i \bar{t}_i \geq \frac{\bar{s}^T \bar{t}}{61m}$. It follows that there is some $\widetilde{M} > 0$ such that $\bar{s}_i, \bar{t}_i \in [\frac{1}{\widetilde{M}}, \widetilde{M}]$ for all $i \in \{1, \dots, m\}$. Since x^k is bounded and s^k, t^k converge, we may assume, without loss of generality,

$$(4.7) \quad s_i^k, t_i^k \in \left[\frac{1}{2\widetilde{M}}, 2\widetilde{M} \right], \quad \|x^k\| \leq \widetilde{M} \quad \text{for all } k.$$

On the compact set (4.7), the Jacobian in (2.4) is nonsingular, and hence its inverse is bounded. Boundedness of the right-hand side in (2.4) implies a uniform bound on the corrections,

$$\|(\Delta x^k, \Delta s^k, \Delta t^k)\| \leq \widehat{M}$$

for some fixed $\widehat{M} > 0$. Hence, and by (4.7), there is some $\epsilon_2 > 0$ such that $\bar{\lambda}_s, \bar{\lambda}_t \geq \epsilon_2$ for all iterations k .

If the safeguard in Step 4 of the line search is active, then $\sigma_k = 3/4$, and from the third block row of (2.4) with $r = \frac{3}{4} \frac{s^T t}{m}$ follows $s^T \Delta t + t^T \Delta s = -\frac{1}{4} s^T t$. Hence,

$$(4.8) \quad \frac{(s + \rho \Delta s)^T (t + \rho \Delta t)}{s^T t} = 1 - \rho/4 + \rho^2 \frac{\Delta s^T \Delta t}{s^T t},$$

while

$$(4.9) \quad \frac{\delta(\rho)}{\delta(0)} = 1 + \rho \frac{\delta'(0)}{\delta(0)} + \rho^2 \frac{\delta''(\xi)}{\delta(0)}.$$

Here,

$$\delta(\rho) = \left(\frac{1}{\|c\|^2} \|c + \sum (t_i + \rho \Delta t_i) \nabla g_i(x + \rho \Delta x)\|^2 + \frac{1}{(1 + \|g(x^0)\|)^2} \|g(x + \Delta x) + s + \rho \Delta s\|^2 \right)^{1/2}$$

and ξ is some intermediate point $\xi \in (0, \rho)$. Note that $\delta'(0) = -\delta(0)$ by the definition of $(\Delta x, \Delta s, \Delta t)$ in (2.4). The denominator $\delta^{3/2}(\xi)$ of the second derivative $\delta''(\xi)$ is bounded below by $\epsilon_1^{3/2}$, and the numerator is uniformly bounded by continuity of g and its derivatives, and the bounds on the iterates and the search directions. Hence, the coefficients of the ρ^2 -terms in (4.8) and (4.9) are bounded and there exists $\epsilon_3 > 0$ independent of k with $\check{\rho} \geq \epsilon_3$.

From C^3 -smoothness of the constraints also follows that there is some $\epsilon_4 > 0$ such that $\hat{\rho} \geq \epsilon_4$ and $\bar{\sigma} \geq \epsilon_4$; both numbers equal 1 if g_i are linear. If $\delta(0) \geq \epsilon_1$ the existence of a uniform lower bound $\hat{\rho} > \epsilon_4 > 0$ follows from the same argument as

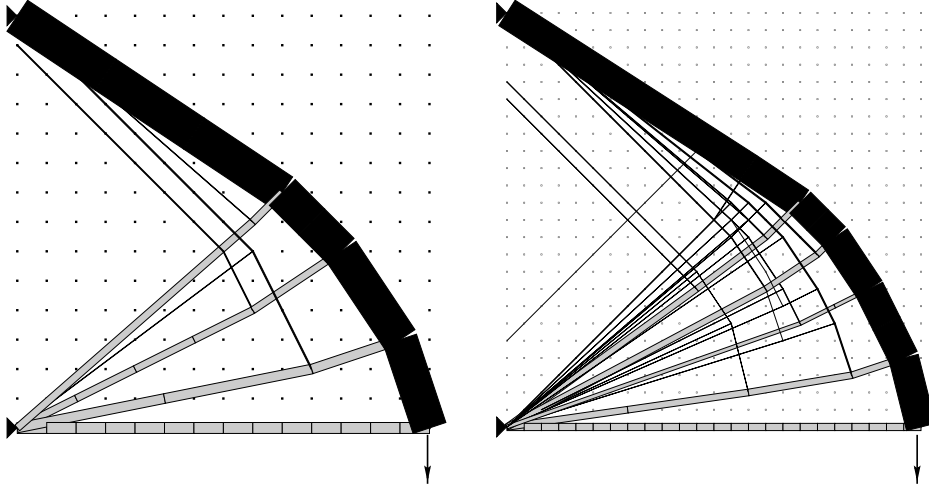


FIG. 5.1. Optimal topologies for the smallest and largest example.

used in (4.9). Since the line search allows a “moderate” increase in the norm of the residuals, the analogous bound follows when $\delta(0) < \epsilon_1$.

Finally, $(1 - \sigma_k) \geq 1/4$, and analogously to condition (12) in [15], there exists $\epsilon_5 > 0$ such that $\tilde{\rho}$ in (4.1) satisfies $\tilde{\rho} > \epsilon_5$. Likewise, since $\sigma_k = 1/2$ (bounded away from zero) when $\beta > 30$, it follows as for condition (13a) in [15] that $\bar{\rho} > \epsilon_5$ as well.

Summarizing, independently of k , the step taken along $(\Delta x^k, \Delta s^k, \Delta t^k)$ has length at least $\min\{\epsilon_2, \epsilon_3, \epsilon_4, \epsilon_5\} > 0$. Thus, since $\sigma_k \leq 3/4$, (4.1) implies that $s^T t$ is reduced by a fixed fraction (independent of k) at each iteration in contradiction to $\bar{s}^T \bar{t} = \epsilon_1 > 0$.

Hence, $\mathbf{d}(\bar{x}, \bar{s}, \bar{t}) = 0$, and $(\bar{x}, \bar{s}, \bar{t})$ satisfies the first-order sufficient conditions for optimality. $\square$

Remark. The above proof of convergence extends in a straightforward manner to the case when Mehrotra’s corrector is used for the complementarity condition only (as suggested by Andersen and Ye [2]), and a safeguard is used to guarantee the line search conditions in section 2.4. The extension of the preceding rules to the Mehrotra corrector as used in our program is not possible. As pointed out, the Mehrotra corrector may locally increase the residuals res_1 and res_2 , and $\hat{\rho}$ may be zero. In the program for the numerical experiments below we used the Mehrotra corrector whenever it resulted in a reduction of $\mathbf{d}$ without violating the constraint $\beta \leq 60$. Else we reverted to a plain primal-dual method as analyzed above. In such very rare cases where the plain primal-dual method was used, however, convergence became slow.

Not all the rules of the line search in section 2.4 are needed to establish convergence. For example, the bounds based on $\bar{\sigma}$ or the restriction in Step 2 of the line search only complicate the above proof but are never really used. In practice, these rules help prevent that the bounds in Steps 3 and 4 become activated.

5. Numerical experiments. An implementation based on Algorithm 2.1 was tested by us on a number of examples derived from the design of cantilever trusses. We consider examples with the same geometry and physical data and with regularly increasing numbers of nodes (unknowns) and potential bars (constraints). In Figure 5.1 we show the optimal topologies for our smallest and largest example. The

TABLE 5.1

Example n	m	Ben-Tal – Zibulevsky				Algorithm 2.1			
		CPU	CPU-FAC	ITER	NEWT	CPU	CPU-FAC	ITER	FAC
450	15556	140		18	36	31		18	4.5
578	25456	262	1.9	16	31	86	2.8	23	3.0
722	39724	677	2.6	18	44	220	2.6	31	3.1
882	59456	1235	1.8	17	46	384	1.7	29	3.2
1058	85252	2390	1.9	19	50	817	2.1	36	2.9
1250	119040	4387	1.8	18	63	1410	1.7	38	3.1
1458	161932	6840	1.6	20	60	2580	1.8	46	2.7
1682	215136					4203	1.6	51	
1922	280916					7497	1.8	54	
2178	361328					13400	1.8	66	

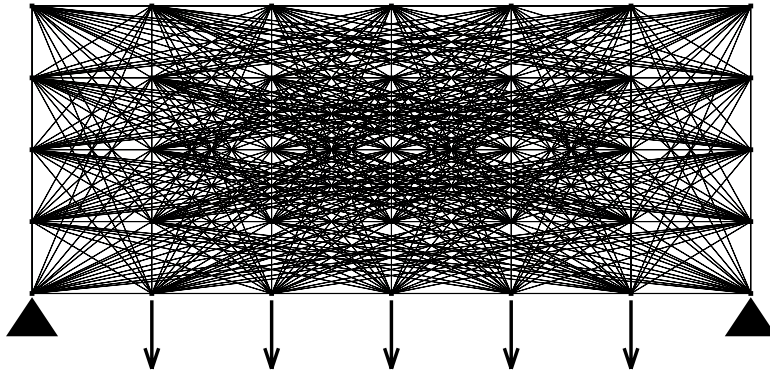
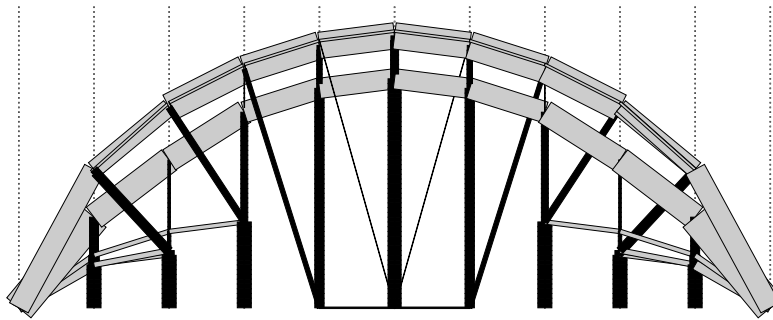
growth of the number of iterations and the growth in computation time is monitored as a function of the number of constraints and number of unknowns.

We compare our results based on Algorithm 2.1 with those obtained by a penalty/barrier algorithm due to Ben-Tal and Zibulevsky [5]. The results are summarized in Table 5.1. In the first two columns we see the data of the examples: number n of unknowns and number m of quadratic constraints. The next columns contain the results obtained by the code due to Ben-Tal and Zibulevsky and by Algorithm 2.1. Column CPU refers to the CPU time (in seconds) on a workstation HP9000-735. CPU-FAC is the ratio of CPU times for two subsequent examples. ITER refers to the number of iterations while NEWT refers to the global number of Newton steps. Recall that in Algorithm 2.1 these two numbers are equal. Let us add that the work for a Newton step is about the same in both algorithms. The last column in Table 5.1 shows the ratio of CPU times for the two algorithms. We remark that the comparison of CPU times should not be taken too seriously since both algorithms are in a “research” stage and are not tuned with respect to CPU time. The last three examples were only computed with Algorithm 2.1.

The last example with about 361328 nonlinear constraints is certainly large scale. We stress again that, even though the test problems of this paper can be reformulated as linear programs, our algorithm did not use such a reformulation but entirely relied on techniques devised for nonlinear programs.

The cantilever truss seems to be “difficult” for both codes, which might be due to the distribution of forces and boundary conditions. We have computed another series of examples for a structure carrying a bridge, which is fixed at its left and right end-points and whose street-nodes are under vertical loads. Figure 5.2 shows as an example a ground-structure mesh of 7×5 nodes and 386 bars. Note that we do not make any a priori assumptions on the shape of the bridge; the shape itself should be the outcome of the optimization process. For this type of truss, the ratio between the two codes was about the same as for cantilever truss. For example, for a ground-structure with $n = 506$ and $m = 19500$, the code due to Ben-Tal and Zibulevsky needed 167 seconds, 15 iterations, and 31 Newton steps, while Algorithm 2.1 needed 16 iterations in 41 seconds. And for the largest example we have computed ($n = 1408, m = 150031$) Algorithm 2.1 needed only 20 iterations (compare to 46 iterations for the cantilever truss of about the same dimension!). The optimal design for the largest example is shown in Figure 5.3. Obviously, an “arch” seems to be the optimal structure for carrying the street.

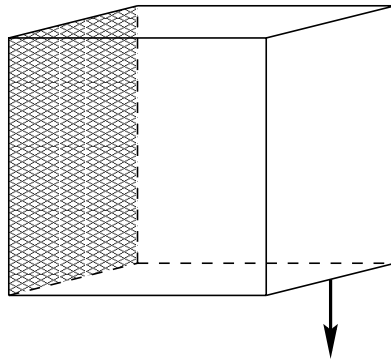
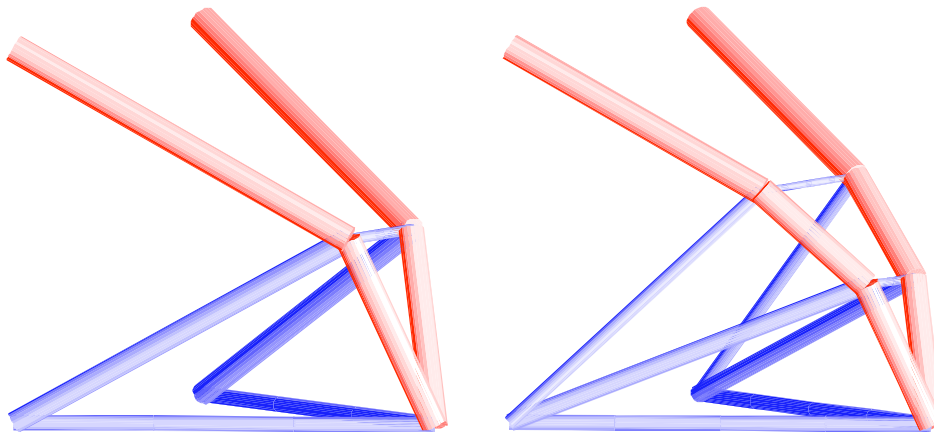
Finally, we present two results on design of three-dimensional cantilever trusses. Figure 5.4 shows the initial data: the geometry (a cube), the load, and the fixed nodes

FIG. 5.2. *Initial layout for a bridge fixed at its end-points.*FIG. 5.3. *Bridge fixed at its end-points.*

(on the shaded surface). Obviously, the complexity of three-dimensional problems is much higher. Table 5.2 presents the size, the number of iterations, and CPU time for two examples with $7 \times 7 \times 7$ and $9 \times 9 \times 9$ nodes, respectively. The optimal structures are depicted in Figure 5.5. Note that the size (number of variables) of the second example is about the same as the size of the largest two-dimensional example (Table 5.1). However, the $9 \times 9 \times 9$ “discretization” is much coarser than even the smallest two-dimensional example (15×15); cf. Figures 5.1 and 5.5. This explains the small number of iterations needed to solve the (large) three-dimensional examples: for a fine grid, there are many nearly optimal designs, and the interior-point method needs more iterations to identify the true one. If (for a coarse mesh) all other design candidates are far from optimality, then the interior-point method will find out and converges quadratically near the optimal solution.

TABLE 5.2

Example		Algorithm 2.1	
n	m	CPU	ITER
1029	48426	450	18
2187	220280	3131	16

FIG. 5.4. *Initial data for three-dimensional cantilever truss.*FIG. 5.5. *Optimal designs for two three-dimensional cantilever trusses.*

Acknowledgments. The authors would like to thank Aharon Ben-Tal and Michael Zibulevsky from Technion, Israel, for making their code available to them.

Part of this work was done while the first author was visiting the Institute of Statistical Mathematics and the Tokyo Institute of Technology in Tokyo, Japan, and he would like to thank Professor Shinji Mizuno and Professor Kojima for their warm hospitality and Professor Takashi Tsuchiya for many helpful discussions about this article.

REFERENCES

- [1] W. ACHTZIGER, M. P. BENDSØE, A. BEN-TAL, AND J. ZOWE, *Equivalent displacement based formulations for maximum strength truss topology design*, Impact Comput. Sci. Engrg., 4 (1992), pp. 315–345.
- [2] E. ANDERSEN AND Y. YE, *On a Homogeneous Algorithm for a Monotone Complementarity Problem with Nonlinear Equality Constraints*, Working paper, Department of Management Sciences, The University of Iowa, Iowa City, IA, January, 1996.
- [3] A. BEN-TAL AND M. P. BENDSØE, *A new method for optimal truss topology design*, SIAM J. Optim., 3 (1993), pp. 322–358.

- [4] A. BEN-TAL AND G. ROTH, *A truncated log barrier algorithm for large scale convex programming and minimax problems implementation and computational results*, Optimization Methods Software, 6 (1996), pp. 283–312.
- [5] A. BEN-TAL AND M. ZIBULEVSKY, *Penalty/barrier multiplier methods for convex programming problems*, SIAM J. Optim., 7 (1997), pp. 347–366.
- [6] E. BLUM AND W. OETTLI, *Mathematische Optimierung*, Springer-Verlag, Berlin, Germany, 1975.
- [7] R. BYRD, M. B. HRIBAR, AND J. NOCEDAL, *An Interior Point Algorithm for Large Scale Non-linear Programming*, Report OTC 97/05, Optimization Technology Center, Northwestern University, Evanston, IL, July, 1997.
- [8] W. S. DORN, R. E. GOMORY, AND H. J. GREENBERG, *Automatic design of optimal structures*, J. Mec., 3 (1964), pp. 25–52.
- [9] R. W. FREUND AND F. JARRE, *A QMR-based interior-point algorithm for solving linear programs*, Math. Programming Ser. B, 76 (1996), pp. 183–210.
- [10] D. DEN HERTOG, *Interior Point Approach to Linear, Quadratic and Convex Programming*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994.
- [11] F. JARRE, *On the convergence of the method of analytic centers when applied to convex quadratic programs*, Math. Programming, 49 (1991), pp. 341–358.
- [12] F. JARRE, *The Method of Analytic Centers for Smooth Convex Programs*, dissertation, Universität Würzburg, Würzburg, Germany, 1989.
- [13] F. JARRE, *Interior-Point Methods via Self-Concordance or Relative Lipschitz Condition*, Habilitation thesis, Universität Würzburg, Würzburg, Germany, 1994.
- [14] M. KOČVARA AND J. ZOWE, *How mathematics can help in design of mechanical structures*, in Proc. of the 16th Biennial Conf. on Numer. Analysis, D. Griffiths and G. Watson, eds., Longman Scientific and Technical, Harlow, 1996, pp. 76–93.
- [15] M. KOJIMA, N. MEGIDDO, AND S. MIZUNO, *A primal-dual infeasible-interior-point algorithm for linear programming*, Math. Programming, 61 (1993), pp. 263–280.
- [16] M. KOJIMA, S. MIZUNO, AND A. YOSHISE, *A primal-dual interior point algorithm for linear programming*, Progr. Math. Programming, Interior-Point and Related Methods, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 29–47.
- [17] I. J. LUSTIG, R. E. MARSTEN, AND D. F. SHANNO, *Computational experience with a primal-dual interior point method for linear programming*, Linear Algebra Appl., 152 (1991), pp. 191–222.
- [18] I. J. LUSTIG, R. E. MARSTEN, AND D. F. SHANNO, *On implementing Mehrotra's predictor-corrector interior-point method for linear programming*, SIAM J. Optim., 2 (1992), pp. 435–449.
- [19] N. MEGIDDO, *Pathways to the optimal set in linear programming*, Progr. Math. Programming, Interior-Point and Related Methods, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 131–158.
- [20] S. MEHROTRA, *On the Implementation of a (Primal-Dual) Interior-Point Method*, Technical report 90-03, Dept. of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL, 1990.
- [21] S. MEHROTRA AND J. SUN, *An interior-point algorithm for solving smooth convex programs based on Newton's method*, in Mathematical Developments Arising from Linear Programming, Contemp. Math. 114, Amer. Math. Soc., Providence, RI, 1990, pp. 265–284.
- [22] R. D. C. MONTEIRO AND I. ADLER, *Interior path following primal-dual algorithms, Part I: Linear programming*, Math. Programming, 44 (1989), pp. 27–41.
- [23] Y. NESTEROV AND A. NEMIROVSKII, *Interior-Point Polynomial Methods in Convex Programming*, SIAM, Philadelphia, 1994.
- [24] Y. NESTEROV AND M. J. TODD, *Self-scaled cones and interior-point methods in nonlinear programming*, Math. Oper. Res., 22 (1997), pp. 1–42.
- [25] D. SHANNO, *private communication*.
- [26] K. TANABE, *Centered Newton method for mathematical programming*, in System Modelling and Optimization, M. Iri and K. Yajima, eds., Springer-Verlag, New York, 1988, pp. 197–206.
- [27] T. WANG, R. D. C. MONTEIRO, AND J.-S. PANG, *An interior point potential reduction method for constrained equations*, Math. Programming, 74 (1996) pp. 159–195.

A NONLINEAR ANALYTIC CENTER CUTTING PLANE METHOD FOR A CLASS OF CONVEX PROGRAMMING PROBLEMS*

F. SHARIFI MOKHTARIAN[†] AND J. L. GOFFIN[‡]

Abstract. A cutting plane algorithm for minimizing a convex function subject to constraints defined by a separation oracle is presented. The algorithm is based on approximate analytic centers. The nonlinearity of the objective function is taken into account, yet the feasible region is approximated by a containing polytope. This containing polytope is regularly updated by adding a new cut through a test point. Each test point is an approximate analytic center of the intersection of a containing polytope and a level set of the nonlinear objective function. We establish the complexity of the algorithm. Our complexity estimate is given in terms of the problem dimension, the desired accuracy of an approximate solution, and other parameters that depend on the geometry of a specific instance of the problem.

Key words. convex programming, interior-point methods, analytic center, cutting planes, potential function, self-concordance

AMS subject classifications. 90C06, 90C25

PII. 51052623496311880

1. Introduction. Recently, many cutting plane or column generation algorithms based on analytic centers have been studied. Sonnevend [22] first introduced the concept of an analytic center of a system of convex inequalities. Goffin, Haurie, and Vial [11], and Ye [25] proposed a cutting plane algorithm that uses the approximate analytic center of the system of inequalities generated in the previous iterations to add a new cut. This method performs well in practice [3, 4, 10, 11, 12]. Later, Goffin, Luo, and Ye [13, 14] analyzed the complexity of this analytic center cutting plane method for solving a convex feasibility problem defined by a separation oracle. Goffin and Vial [15] have extended this result by considering shallow, deep, and very deep cuts. Complexity bounds have also been established for several related methods by Nesterov [19], Atkinson and Vaidya [2], Kiwiel [16], and Altman and Kiwiel [1]. Extensions to the case of multiple cuts are given by Ye [26]. Luo and Sun [17] have used an analytic center-based column-generation algorithm for a convex quadratic feasibility problem that uses quadratic cuts. They have also extended the case to self-concordant convex inequalities using convex cuts [18]. Interior-point and cutting plane methods have also been combined in a different way by Den Hertog et al. [6]; they proposed a logarithmic barrier cutting plane method for convex (possibly non-smooth, semi-infinite) programming and provided some computational results. Further computational results for the same method were later given by Kaliski et al. [7]. Although this method is of interior-point nature, it is not an analytic center cutting plane method.

*Received by the editors November 11, 1996; accepted for publication (in revised form) September 2, 1997; published electronically September 23, 1998. This research was supported by the Natural Sciences and Engineering Research Council of Canada, grant OPG0004152, and by the FCAR of Quebec.

<http://www.siam.org/journals/siopt/8-4/31188.html>

[†]Department of Mathematics and Statistics, McGill University, 805 Sherbrooke Street West, Montreal, Quebec, Canada H3A 2K6 (faranak@math.mcgill.ca).

[‡]Faculty of Management, McGill University, 1001 Sherbrooke Street West, Montreal, Quebec, Canada H3A 1G5 (goffin@management.mcgill.ca).

In this paper, we consider the problem

$$(CP) \begin{cases} \min & f(y), \\ & y \in \Gamma, \end{cases}$$

where $f : [0, 1]^m \rightarrow R$ is convex and C^3 -smooth and $\Gamma \subset R^m$ is a closed convex set with a nonempty interior contained in the cube $\Omega^0 = [0, 1]^m$. The set Γ is defined implicitly by a *separation oracle* which for every $\bar{y} \in \Omega^0$ either answers that $\bar{y}$ belongs to the interior of Γ or generates a cut $a^T y \leq a^T \bar{y}$ with the property that $\{y \in R^m : a^T y \leq a^T \bar{y}\} \supset \Gamma$. Without loss of generality, we assume that a is normalized so that $\|a\| = 1$. The set Γ can be defined by, e.g., a finite or infinite number of pseudoconvex constraints, a finite or infinite system of convex inequalities where some or all of the constraints are nonsmooth, or by an infinite or large number of linear inequalities. We also make the following assumption.

Assumption 1. For any z (an upper bound on the objective function),

$$\varphi_z(y) = -\ln(z - f(y))$$

is α -self-concordant on $(0, 1)^m \cap \{y \in R^m : f(y) < z\}$, i.e., for all $h \in R^m$ and for all $y \in (0, 1)^m \cap \{y \in R^m : f(y) < z\}$,

$$\{D^3 \varphi_z(y)[h, h, h]\}^2 \leq 4\alpha(h^T \nabla^2 \varphi_z(y)h)^3,$$

where $D^3 \varphi_z(y)[h, h, h]$ denotes the third differential of φ_z at y and h .

We also assume without loss of generality that $\alpha \geq 1$. The assumption that α is independent of z is not restrictive, for we can conclude, using Proposition 2.1.1 (ii) of Nesterov and Nemirovskii [20], that if $\psi(y, z) = -\ln(z - f(y))$ is α -self-concordant on $\{(y, z) \in R^{m+1} : y \in (0, 1)^m, z > f(y)\}$, then φ_z is α -self-concordant on $(0, 1)^m \cap \{y \in R^m : f(y) < z\}$. Problems formulated as (CP) appear a lot in practice. For example, some multicommodity flow problems give rise to very large nonlinear programs with linear constraints [10].

We consider an analytic center-based cutting plane algorithm for solving (CP) that takes the nonlinearity of the objective function into account. The algorithm starts with $\Omega^0 = [0, 1]^m$ as the original outer approximation of the feasible region. We assume that the analytic center of Ω^0 is in the interior of Γ . As we will discuss later on, this is not a restrictive assumption and makes the analysis simpler. Using this point, we find an initial upper bound on the optimal value, $z(0)$. Then the nonlinear cut $f(y) \leq z(0)$ corresponding to the objective is added. This defines a “set of localization” that contains the solution set to (CP) .

The algorithm consists of inner and outer iterations. At each inner iteration, an approximate analytic center of the set of localization is calculated and tested by the oracle for feasibility. If the test point is not feasible, a linear cut will be added that goes through the test point. The inner iterations continue until a feasible test point is found. If the feasible test point is an approximate solution of (CP) , the algorithm stops. Otherwise, an outer iteration is performed in which the value of the upper bound on the objective is reduced. As the number of iterations increases, the set of localization shrinks and the algorithm eventually finds an approximate solution of the problem. Assumption 1, in view of the fact that the feasibility cuts added in the algorithm are all linear, implies that the parameter of self-concordance of the potential function for any set of localization remains equal to α (see, e.g., Nesterov and Nemirovskii [20, Proposition 2.1.1]). The self-concordancy property plays an important role in the complexity analysis of our algorithm.

Our algorithm can be thought of as an extension of the central cutting plane algorithm of Goffin, Luo, and Ye [14] for the convex feasibility problem. However, we consider the problem in the form of an optimization problem involving a nonlinear objective. The important feature of this extension is that a nonlinear cut corresponding to a level surface of the objective is considered in our case, whereas only linear cuts are used in their algorithm.

The central cutting surface algorithm of Luo and Sun [18], on the other hand, uses convex cuts, yet the feasible set that they consider is defined by a finite number of convex differentiable inequalities that admit a self-concordant logarithmic barrier. They show that their algorithm has a polynomial worst case complexity in terms of the total number of constraints and the maximum common slack of the original inequality system. In our case, however, the feasible region is implicitly defined by a separation oracle, and in this sense our case considers a broader class of problems. A challenge in our case is how to handle the complexity analysis when there is no limit on the number of cuts the oracle can produce. Furthermore, Luo and Sun [18] consider the feasibility version of the problem, whereas we consider the problem in the context of optimization. Our complexity estimate is given in terms of the dimension of the problem, the accuracy of an approximate optimal solution, and other parameters depending on the geometry of a specific instance of the problem. They also add their nonlinear cuts as “shallow” cuts. A new cut added in their analysis must be shallow enough so that the previous test point (an approximate analytic center of the previous set of localization) remains within the region of quadratic convergence of Newton’s method, used to calculate an approximate analytic center of the augmented set. In our analysis, on the other hand, the feasibility cuts are added through the approximate analytic center of the set of localization, and there is no restriction on how shallow a particular translation of the nonlinear objective cut must be. However, as it will be seen in this paper, some of the results of Luo and Sun [18], i.e., changes in the potential after the addition or translation of a nonlinear cut, play important roles in our analysis as well. The other important key element in the complexity analysis of our algorithm is a result due to Nesterov [19] regarding the change in the potential after the addition of a linear cut to the set of localization.

This paper is organized as follows. In section 2, we review some basic concepts and results to be used in the subsequent analysis. In section 3, we give our cutting plane algorithm. In section 4, some results needed for complexity of the algorithm will be given. Finally, complexity of the algorithm will be discussed in sections 5 and 6.

We will adopt the standard notation used in the interior point literature. For any generic vector u , u^T will denote the vector transpose, and the corresponding capital letter U will denote the diagonal matrix whose i th diagonal element is given by the i th component of u . The vector $\vec{e}$ will represent the vector of ones of appropriate dimension. For a real number a , $\lceil a \rceil$ represents the smallest integer greater than or equal to a . The inradius of a compact convex set G in R^m denoted by $\rho(G)$ is the radius of the largest sphere contained in G . The interior of G will be represented by G_{int} . For two symmetric $m \times m$ -matrices A and B we write $A \succeq B$ if the matrix $A - B$ is positive semidefinite. We will also use the following notation:

$$\|u\|_H = \langle u, Hu \rangle^{1/2},$$

$$\|u\|_H^* = \langle u, H^{-1}u \rangle^{1/2},$$

where H is a given symmetric positive definite matrix and $\langle \cdot, \cdot \rangle$ represents the standard inner product in R^m . The norm $\| \cdot \|$ simply represents the Euclidean norm. Throughout this paper, we assume that f_* is the optimal value of (CP) .

2. Preliminaries. Let Ω be a bounded set defined as follows:

$$(2.1) \quad \Omega = \{y \in R^m : f(y) \leq z, a_i^T y \leq c_i, i = 1, \dots, n\},$$

where f is the objective function in (CP) and satisfies Assumption 1. Suppose that Ω_{int} is nonempty. The potential function of Ω is defined as

$$(2.2) \quad \phi(\Omega, y) = - \sum_{i=1}^n \ln(c_i - a_i^T y) - \ln(z - f(y)).$$

Note that $\phi(\Omega, y)$ is α -self-concordant. We recall (see [20, Definition 2.1.1]) that a function is strongly α -self-concordant on Ω_{int} if it is α -self-concordant on Ω_{int} and, in addition, tends to infinity along every sequence $\{y_i \in \Omega_{int}\}$ converging to a boundary point of Ω_{int} . Hence, by its definition, $\phi(\Omega, y)$ will be automatically strongly α -self-concordant since it is α -self-concordant. The gradient and the Hessian of $\phi(\Omega, y)$ are given by

$$g(\Omega, y) = \nabla \phi(\Omega, y) = \frac{\nabla f(y)}{z - f(y)} + \sum_{i=1}^n \frac{a_i}{c_i - a_i^T y}$$

and

$$H(\Omega, y) = \nabla^2 \phi(\Omega, y) = \frac{\nabla f(y) \nabla f(y)^T}{(z - f(y))^2} + \frac{\nabla^2 f(y)}{z - f(y)} + \sum_{i=1}^n \frac{a_i a_i^T}{(c_i - a_i^T y)^2}.$$

The Newton direction of $\phi(\Omega, y)$ at y will be given by

$$q(\Omega, y) = -H(\Omega, y)^{-1} g(\Omega, y).$$

The min-potential of Ω is defined as

$$(2.3) \quad P(\Omega) = \min_{y \in \Omega_{int}} \phi(\Omega, y).$$

The unique point in Ω_{int} where this minimum is attained is called the “analytic center” of Ω . Let y^a be this analytic center.

We recall from [20] that one proximity measure between the analytic center y^a and a given $y \in \Omega_{int}$ is

$$\sqrt{\alpha} \|g(\Omega, y)\|_{H(\Omega, y)}^*.$$

Note that

$$\|q(\Omega, y)\|_{H(\Omega, y)} = \|g(\Omega, y)\|_{H(\Omega, y)}^*.$$

Another measure is

$$\sqrt{\alpha} \|y - y^a\|_{H(\Omega, y)}.$$

Finally, the “gap” $\phi(\Omega, y) - P(\Omega)$ is a third measure of proximity.

DEFINITION 2.1. Let $0 < \beta < 1$. We say $y \in \Omega_{int}$ is a β -approximate analytic center of Ω if

$$\sqrt{\alpha} \|g(\Omega, y)\|_{H(\Omega, y)}^* \leq \beta.$$

The next two lemmas relate the given proximity measures.

LEMMA 2.2. Let $y \in \Omega_{int}$ and $\gamma \in [0, (\sqrt{2} - 1)^2]$. If y is a γ -approximate analytic center of Ω , then

$$\sqrt{\alpha} \|y - y^a\|_{H(\Omega, y)} \leq \frac{\gamma}{(2 - \sqrt{2})}.$$

Proof. See Nesterov [19, Corollary 2.1]. $\square$

LEMMA 2.3. Let $y \in \Omega_{int}$. If y is a $(1/3)$ -approximate analytic center of Ω , then

$$\phi(\Omega, y) - P(\Omega) \leq \frac{(\|g(\Omega, y)\|_{H(\Omega, y)}^*)^2}{1 - \{\frac{9}{4}\sqrt{\alpha}\|g(\Omega, y)\|_{H(\Omega, y)}^*\}^2}.$$

Proof. See Den Hertog [5, Lemma 3.14]. $\square$

The following lemma is due to Nesterov and Nemirovskii [20, Proposition 2.2.2] and states that a sufficient decrease in the potential function can be obtained by taking a step of suitable size along the Newton direction.

LEMMA 2.4. Suppose that $y \in \Omega_{int}$ and

$$\sigma = 1/\{1 + \sqrt{\alpha}\|g(\Omega, y)\|_{H(\Omega, y)}^*\}.$$

Then $y^+ = y + \sigma q(\Omega, y) \in \Omega_{int}$ and

$$\phi(\Omega, y) - \phi(\Omega, y^+) \geq \{\sqrt{\alpha}\|g(\Omega, y)\|_{H(\Omega, y)}^* - \ln(1 + \sqrt{\alpha}\|g(\Omega, y)\|_{H(\Omega, y)}^*)\}/\alpha.$$

The following result is part of Theorem 2.2.3 of [20].

LEMMA 2.5. Suppose that $y \in \Omega_{int}$, and let

$$(2.4) \quad \gamma_* = 2 - \sqrt{3}.$$

If $\sqrt{\alpha}\|g(\Omega, y)\|_{H(\Omega, y)}^* < \gamma_*$, then $y^+ = y + q(\Omega, y) \in \Omega_{int}$ and

$$(2.5) \quad \sqrt{\alpha}\|g(\Omega, y^+)\|_{H(\Omega, y^+)}^* \leq \left\{ \frac{\sqrt{\alpha}\|g(\Omega, y)\|_{H(\Omega, y)}^*}{1 - \sqrt{\alpha}\|g(\Omega, y)\|_{H(\Omega, y)}^*} \right\}^2.$$

Lemma 2.5 implies that if $y \in \Omega_{int}$ satisfies

$$0 < \beta < \sqrt{\alpha}\|g(\Omega, y)\|_{H(\Omega, y)}^* < \gamma_*,$$

then we can take full Newton steps to calculate a β -approximate analytic center of Ω . We note that the quantities $\sqrt{\alpha}\|g(\Omega, y^i)\|_{H(\Omega, y^i)}^*$, $i = 0, 1, \dots$, with $y^0 = y$, by (2.5), decrease quadratically. Therefore, the total number of Newton iterations to find a β -approximate analytic center of Ω , starting at y , is at most

$$(2.6) \quad \bar{N} = \left\lceil \frac{1}{\ln 2} \ln \left(\frac{\ln \beta - 2 \ln(1 - \gamma_*)}{\ln \gamma_* - 2 \ln(1 - \gamma_*)} \right) \right\rceil,$$

an $O(1)$ number depending on β only.

DEFINITION 2.6. A mapping T_u from the convex subset U of a linear space onto the class of subsets of R^m is called a concave array if

$$u_1, u_2 \in U, \quad \lambda \in [0, 1]$$

implies

$$T_{\lambda u_1 + (1-\lambda)u_2} \supset \lambda T_{u_1} + (1-\lambda)T_{u_2}.$$

Let

$$(2.7) \quad \Gamma_\xi = \{y \in \Gamma : f(y) \leq f_* + \xi\},$$

where $\xi \geq 0$.

LEMMA 2.7. The mapping Γ_ξ is a concave array on R_+ .

Proof. The proof is elementary and follows from the convexity of f , Γ , and the definition of a concave array. $\square$

For more information on concave arrays and behavior of level sets of a convex function, we refer the reader to Eggleston [8] and Goffin [9].

THEOREM 2.8. Let $0 \leq \lambda \leq 1$. Then

$$\rho(\Gamma_{\lambda\xi}) \geq \lambda\rho(\Gamma_\xi).$$

Proof. By Lemma 2.7, we have

$$\Gamma_{\lambda\xi} \supset \lambda\Gamma_\xi + (1-\lambda)\Gamma_0.$$

Let y^* be any optimal solution of (CP). Then

$$\Gamma_{\lambda\xi} - y^* \supset \lambda(\Gamma_\xi - y^*) + (1-\lambda)(\Gamma_0 - y^*).$$

Now $0 \in (1-\lambda)(\Gamma_0 - y^*)$. Hence

$$\lambda(\Gamma_\xi - y^*) + (1-\lambda)(\Gamma_0 - y^*) \supset \lambda(\Gamma_\xi - y^*).$$

This implies that

$$\Gamma_{\lambda\xi} - y^* \supset \lambda(\Gamma_\xi - y^*)$$

or

$$\rho(\Gamma_{\lambda\xi} - y^*) \geq \rho(\lambda(\Gamma_\xi - y^*)) \geq \lambda\rho(\Gamma_\xi - y^*).$$

Using the trivial equality

$$\rho(G) = \rho(G - t),$$

which holds for any compact convex set $G \in R^m$ and any vector $t \in R^m$, we obtain

$$\rho(\Gamma_{\lambda\xi}) \geq \lambda\rho(\Gamma_\xi). \quad \square$$

COROLLARY 2.9. Let $w^0 \in \Gamma$, $z(0) = f(w^0)$, and $\epsilon > 0$ be a given tolerance. Suppose $z(0) - f_* \geq \epsilon$. Then

$$\frac{\rho(\Gamma_\epsilon)}{\epsilon} \geq \frac{\rho(\Gamma_{z(0)-f_*})}{z(0) - f_*}.$$

Proof. In Theorem 2.8, let $\xi = z(0) - f_*$ and $\lambda\xi = \epsilon$. $\square$

COROLLARY 2.10. Let $0 < \epsilon \leq 1$. Then

$$\frac{\rho(\Gamma_\epsilon)}{\epsilon} \geq \rho(\Gamma_1).$$

Proof. In Theorem 2.8, let $\xi = 1$ and $\lambda = \epsilon$. $\square$

We will make the following final assumption throughout this paper.

Assumption 2. The interior of Γ_1 contains a full-dimensional closed ball of radius $0 < \delta < 1/2$.

We assume that δ in Assumption 2 is a known constant. Assumption 2, in view of Corollary 2.10, will be used in the first stopping rule of the algorithm to guarantee that the total number of feasibility cuts produced by the algorithm is bounded.

3. The cutting plane algorithm. In this section, we first explain some notation that will be used in the algorithm and later in its complexity analysis. The algorithm starts with Ω^0 and its analytic center $w^0 = (1/2)\vec{e}$. We assume that $w^0 \in \Gamma_{int}$. This assumption is not restrictive. Indeed, the complexity analysis of the problem of finding an interior point of Γ is studied by Goffin, Luo, and Ye [14]. There is a known bound on the total number of iterations required to find such a point. This means that we can start the algorithm from a given bounded polyhedron and an approximate analytic center of it. We can assume that this polyhedron is Ω^0 to make the analysis simpler. Having w^0 , the algorithm determines an initial upper bound on f_* and adds the nonlinear cut corresponding to the objective, i.e., $f(y) \leq z(0)$ to Ω^0 to get Ω_0^0 . In the algorithm, translations of the nonlinear objective cut will be performed as well. We refer to these translations as outer iterations. For example, the constraint $f(y) \leq z(l)$ represents the level of the objective function in the l th outer iteration. Furthermore, Ω_l^j represents the “set of localization” when the objective constraint is translated for the l th time and for that l , the j th cut is added. The approximate analytic center of Ω_l^j is represented by y_l^j . The maximum number of cuts added during the l th objective translation is represented by $\eta(l)$. The $(j+1)$ th cut added in the l th outer iteration is represented by $(a_l^{j+1})^T y \leq c_l^{j+1}$, where $c_l^{j+1} = (a_l^{j+1})^T y_l^j$. For simplicity, the set $\Omega_l^{\eta(l)}$ is represented by Ω_l^f , and the feasible point found in the l th objective translation which is the approximate analytic center of Ω_l^f is represented by y_l^f . Index $k(l, j)$ represents the total number of feasibility cuts produced until the (l, j) th iteration. Tolerance $\epsilon > 0$ represents the desired accuracy of an approximate solution at termination. Any $y^* \in \Gamma_\epsilon$ will be referred to as an ϵ -solution of (CP). We now describe the algorithm.

The Cutting Plane Algorithm.

Step 0 Select $0 < \epsilon < 1$, $0 < \theta < 1$, $0 < \gamma \leq (\sqrt{2} - 1)^2/2$.

Set $\Omega^0 = [0, 1]^m$, $0 < \delta < 1/2$, $w^0 = (1/2)\vec{e}$, and
 $c(\alpha, \gamma) = 1 - (\sqrt{2} - \ln \sqrt{2} - 1)/\alpha + \ln[(\sqrt{2} - 1)^2 - \gamma]/\sqrt{\alpha}$.

Step 1 Set $\tilde{r} = \|\nabla f(w^0)\|_{H(\Omega^0, w^0)}^*$,

$$\zeta(\alpha, \gamma) = -\ln \tilde{r} - \ln \left(\frac{\sqrt{\alpha}}{\gamma} + \frac{2\gamma}{\sqrt{\alpha}(2 - \sqrt{2})} \right),$$

$$z(0) = f(w^0) + \sqrt{\alpha}\tilde{r}/\gamma,$$

$$\Omega_0^0 = \Omega^0 \cup \{y \in R^m : f(y) \leq z(0)\},$$

$$k(0, 0) = 0, l = 0, j = 0, \text{ and } y_{-1}^f = w^0.$$

Step 2 Find a γ -approximate analytic center of Ω_l^j , y_l^j .

Step 3 Check (by querying the oracle) if $y_l^j \in \Gamma_{int}$.

If $y_l^j \in \Gamma_{int}$ **Then**

set $\eta(l) = j$ and $y_l^f = y_l^j$.

Else the oracle returns the cut $a_l^{j+1}y \leq c_l^{j+1}$ with $\|a_l^{j+1}\| = 1$.

Update $\Omega_l^{j+1} = \Omega_l^j \cup \{y \in R^m : a_l^{j+1}y \leq c_l^{j+1}\}$.

Set $k(l, j+1) = k(l, j) + 1$ and $j := j + 1$.

If (Stopping Criterion 1)

$$\frac{1+4m \ln\left(1+\frac{k(l,j)}{8m^2}\right)}{1+2m+k(l,j)} \exp\left(-\frac{2c(\alpha,\gamma)k(l,j)}{1+2m+k(l,j)} - \frac{2\zeta(\alpha,\gamma)}{1+2m+k(l,j)}\right) \leq \frac{(\epsilon\delta)^2}{2m}$$

Then stop (y_{l-1}^f is an ϵ -solution).

Else return to Step 2.

Endif

Endif

Step 4 If (Stopping Criterion 2)

$$z(l) - f(y_l^f) \leq \frac{4\epsilon}{5(1+2m+k(l,\eta(l)))}$$

Then stop (y_l^f is an ϵ -solution).

Else set

$$z(l+1) = z(l) - \theta(z(l) - f(y_l^f)),$$

$$k(l+1, 0) = k(l, \eta(l)), j = 0,$$

$$l := l + 1, \text{ and return to Step 2.}$$

Endif

End

4. Potential increase. We consider the changes in the min-potential in the following three cases:

1. when the nonlinear objective cut is translated;
2. when the objective cut is added; and
3. when a linear feasibility cut is added.

In this section, we assume that the set Ω is defined by (2.1). Hence $\phi(\Omega, y)$ is strongly α -self-concordant on Ω_{int} . The following lemma relates the slacks of the constraint $f(y) \leq z$ of Ω evaluated at the exact and an approximate analytic center of it.

LEMMA 4.1. *Let y^a and y^c be the exact center and a γ -approximate analytic center of Ω , respectively, where $\gamma \in [0, (\sqrt{2} - 1)^2]$. Then*

$$(4.1) \quad 0 < (z - f(y^a)) \leq \left(1 + \frac{\gamma}{(2 - \sqrt{2})\sqrt{\alpha}}\right) (z - f(y^c)).$$

Proof. Since $\|g(\Omega, y^c)\|_{H(\Omega, y^c)}^* \leq \gamma/\sqrt{\alpha}$, it follows from Lemma 2.2 that

$$\|y^a - y^c\|_{H(\Omega, y^c)} \leq \frac{\gamma}{(2 - \sqrt{2})\sqrt{\alpha}}.$$

But

$$\frac{\nabla f(y^c) \nabla f(y^c)^T}{(z - f(y^c))^2} \preceq H(\Omega, y^c).$$

Therefore,

$$\frac{|\langle \nabla f(y^c), y^a - y^c \rangle|}{z - f(y^c)} \leq \frac{\gamma}{(2 - \sqrt{2})\sqrt{\alpha}}$$

or

$$\begin{aligned} \frac{z - f(y^a)}{z - f(y^c)} - 1 &\leq \frac{|\langle \nabla f(y^c), y^a - y^c \rangle|}{z - f(y^c)} \\ &\leq \frac{\gamma}{(2 - \sqrt{2})\sqrt{\alpha}}, \end{aligned}$$

which implies (4.1). $\square$

COROLLARY 4.2. *Let Ω be updated to Ω_θ as follows:*

$$\Omega_\theta = \{y \in R^m : a_i^T y \leq c_i, i = 1, \dots, n, f(y) \leq z - \theta(z - f(y^c))\},$$

where y^c is a γ -approximate analytic center of Ω with $\gamma \in [0, (\sqrt{2} - 1)^2]$ and $0 < \theta < 1$. Then

$$P(\Omega_\theta) \geq P(\Omega) + \lambda(\alpha, \gamma)\theta,$$

where $\lambda(\alpha, \gamma) = \left(1 + \frac{\gamma}{(2 - \sqrt{2})\sqrt{\alpha}}\right)^{-1}$.

Proof. The proof of Luo and Sun [18, Lemma 3.1] can be used in this case as well, using Lemma 4.1. $\square$

The next lemma is a combination of Lemmas 3.2 and 6.2 of Luo and Sun [18]. It shows the changes in the min-potential when the nonlinear objective cut is added and the proximity of the old approximate analytic center to the new analytic center.

LEMMA 4.3. *Let $0 < \gamma \leq (\sqrt{2} - 1)^2/2$. Consider the set of localization obtained after the addition of the objective cut*

$$\Omega_0^0 = \Omega^0 \cap \{y \in R^m : f(y) \leq f(w^0) + \sqrt{\alpha}\tilde{r}/\gamma\},$$

where $\tilde{r} = \|\nabla f(w^0)\|_{H(\Omega^0, w^0)}^*$. Then

$$(4.2) \quad P(\Omega_0^0) \geq P(\Omega^0) - \ln \tilde{r} - \ln \left(\frac{\sqrt{\alpha}}{\gamma} + \frac{2\gamma}{\sqrt{\alpha}(2 - \sqrt{2})} \right)$$

and

$$(4.3) \quad \|g(\Omega_0^0, w^0)\|_{H(\Omega_0^0, w^0)}^* \leq \|g(\Omega^0, w^0)\|_{H(\Omega^0, w^0)}^* + \frac{\gamma}{\sqrt{\alpha}}.$$

Finally, we consider the changes in the min-potential when a feasibility cut is added. The following lemma is basically Lemma 2.1 of Nesterov [19] applied to our case. There, the result is given for strongly 1-self-concordant functions, yet it can be extended to strongly α -self-concordant functions (our $\phi(\Omega, y)$) as well. The proof is given in the appendix. This lemma is also an extension of Ye [25, Theorem 2], which gives a lower bound on the min-potential for a reduced polyhedron.

LEMMA 4.4. *Let y^c be a γ -approximate analytic center of Ω with*

$$\gamma \in [0, (\sqrt{2} - 1)^2].$$

Consider the new set

$$\Omega^+ = \{y \in R^m : f(y) \leq z, a_i^T y \leq c_i, i = 1, \dots, n, (a_{n+1})^T y \leq (a_{n+1})^T y^c\}$$

with $a_{n+1} \in R^m$ such that $\|a_{n+1}\| = 1$. Let

$$(4.4) \quad y^+ = y^c - \frac{[H(\Omega, y^c)]^{-1} a_{n+1}}{3\sqrt{\alpha} \|a_{n+1}\|_{H(\Omega, y^c)}^*}.$$

Then $y^+ \in (\Omega^+)_{int}$ and

$$(4.5) \quad \begin{aligned} \phi(\Omega, y^c) - \ln \|a_{n+1}\|_{H(\Omega, y^c)}^* + c(\alpha, \gamma) &\leq P(\Omega^+) \leq \phi(\Omega^+, y^+) \\ &\leq \phi(\Omega, y^c) - \ln \|a_{n+1}\|_{H(\Omega, y^c)}^* + \bar{c}(\alpha, \gamma), \end{aligned}$$

where

$$(4.6) \quad c(\alpha, \gamma) = 1 - (\sqrt{2} - \ln \sqrt{2} - 1)/\alpha + \ln[(\sqrt{2} - 1)^2 - \gamma]/\sqrt{\alpha},$$

$$(4.7) \quad \bar{c}(\alpha, \gamma) = \gamma/(3\alpha) + \ln(3\sqrt{\alpha}) - (\ln(2/3) + 1/3)/\alpha.$$

As a consequence of Lemma 4.4 and using the same notation and assumptions, we have the following changes in the min-potential while adding a linear feasibility cut:

$$(4.8) \quad P(\Omega^+) \geq P(\Omega) - \ln \|a_{n+1}\|_{H(\Omega, y^c)}^* + c(\alpha, \gamma).$$

5. Convergence and complexity analysis. In this section, we will analyze the complexity of the cutting plane algorithm. For simplicity, we will use the following notation in this section:

$$\begin{aligned} H_u^v &= \nabla^2 \phi(\Omega_u^v, y_u^v), \\ r_u^v &= \|a_u^{v+1}\|_{H_u^v}^*, \end{aligned}$$

where $u \geq 0$, $v = 0, \dots, \eta(u) - 1$. In the next lemma, we find a lower bound on the min-potential.

LEMMA 5.1. For all $l \geq 0$ and for all $j = 0, \dots, \eta(l)$,

$$P(\Omega_l^j) \geq P(\Omega^0) + \lambda(\alpha, \gamma)\theta l - \sum_{u=0}^{l-1} \sum_{v=0}^{\eta(l)-1} \ln r_u^v - \sum_{v=0}^{j-1} \ln r_l^v + c(\alpha, \gamma)k(l, j) + \zeta(\alpha, \gamma),$$

where $c(\alpha, \gamma)$ and $\zeta(\alpha, \gamma)$ are as defined in Step 0 and Step 1 of the algorithm, and $\lambda(\alpha, \gamma)$ is as defined in Corollary 4.2.

Proof. We discuss the case $l \geq 1$. The case $l = 0$ can be proven very similarly. In view of Corollary 4.2 and (4.8), we have

$$\begin{aligned} P(\Omega_l^j) &\geq P(\Omega_l^{j-1}) - \ln r_l^{j-1} + c(\alpha, \gamma) \\ &\geq P(\Omega_l^0) - \sum_{v=0}^{j-1} \ln r_l^v + c(\alpha, \gamma)j \\ &\geq P(\Omega_{l-1}^f) - \sum_{v=0}^{j-1} \ln r_l^v + \theta\lambda(\alpha, \gamma) + c(\alpha, \gamma)j \\ &\quad (\text{by the translation of the objective cut}) \\ &\geq P(\Omega_0^0) + \lambda(\alpha, \gamma)\theta l - \sum_{v=0}^{j-1} \ln r_l^v - \sum_{u=0}^{l-1} \sum_{v=0}^{\eta(l)-1} \ln r_u^v + c(\alpha, \gamma)k(l, j). \end{aligned}$$

Now, in view of Lemma 4.3 ((4.2)) (applied to the addition of the objective cut) and the definition of $\zeta(\alpha, \gamma)$, we have

$$P(\Omega_l^j) \geq P(\Omega^0) + \lambda(\alpha, \gamma)\theta l - \sum_{v=0}^{j-1} \ln r_l^v - \sum_{u=0}^{l-1} \sum_{v=0}^{\eta(l)-1} \ln r_u^v + c(\alpha, \gamma)k(l, j) + \zeta(\alpha, \gamma). \quad \square$$

Now for fixed $u \geq 0$ and $0 \leq v \leq \eta(u) - 1$ and for any $y \in (\Omega_u^v)_{int}$, we consider the slacks corresponding to the constraints of Ω_u^v at y (excluding the $2m$ constraints corresponding to the unit box $[0, 1]^m$) represented as follows:

$$s_i^r(y) = c_i^r - (a_i^r)^T y,$$

where $1 \leq r \leq \eta(i)$ for $0 \leq i \leq u - 1$ and $1 \leq r \leq v$ for $i = u$. Since Γ is contained in the cube $[0, 1]^m$ and the linear constraints are given by normal vectors of unit norm, we can easily conclude (see Goffin, Luo, and Ye [14, Lemma 6.2]) that for i and r as defined above,

$$(5.1) \quad 0 \leq s_i^r(y) \leq \sqrt{m}.$$

We will now find overestimators for r_u^v for $u \geq 0$, $v = 0, \dots, \eta(u) - 1$ that are easier to manipulate. We, therefore, construct the following matrices, using a construction due to Nesterov [19]:

$$\begin{aligned} B^0 &= 8I, \\ B^{k(l, j+1)} &= B^{k(l, j)} + \frac{a_l^{j+1}(a_l^{j+1})^T}{m}, \quad l \geq 0, \quad 0 \leq j \leq \eta(l) - 1, \\ B^{k(l+1, 0)} &= B^{k(l, \eta(l))}, \quad l \geq 0. \end{aligned}$$

(Note that $k(0, 0) = 0$ and $k(l+1, 0) = k(l, \eta(l))$.)

LEMMA 5.2. *The above matrices satisfy the following:*

$$H_u^v \succeq B^{k(u, v)}, \quad u \geq 0, \quad v = 0, \dots, \eta(u) - 1.$$

Proof. We give the proof for the case $u \geq 1$ and $v = 1, \dots, \eta(u) - 1$. The other case corresponding to $u = 0$ can be proven very similarly:

$$\begin{aligned} H_u^v &= \sum_{i=0}^{u-1} \sum_{r=1}^{\eta(i)} \frac{a_i^r (a_i^r)^T}{[s_i^r(y_u^v)]^2} + \sum_{r=1}^v \frac{a_u^r (a_u^r)^T}{[s_u^r(y_u^v)]^2} + (Y_u^v)^{-2} + (I - Y_u^v)^{-2} \\ &\quad + \frac{\nabla f(y_u^v) \nabla f(y_u^v)^T}{[z(u) - f(y_u^v)]^2} + \frac{\nabla^2 f(y_u^v)}{z(u) - f(y_u^v)} \\ &\succeq \sum_{i=0}^{u-1} \sum_{r=1}^{\eta(i)} \frac{a_i^r (a_i^r)^T}{m} + \sum_{r=1}^v \frac{a_u^r (a_u^r)^T}{m} + 8I \\ &\quad \text{(by (5.1) and convexity of } f) \\ &= B^{k(u, v)}. \end{aligned}$$

The result is proved. $\square$

LEMMA 5.3. *Let*

$$w^{k(u,v)} = \|a_u^{v+1}\|_{B^{k(u,v)}}^*$$

for $u \geq 0$ and for $v = 0, \dots, \eta(u) - 1$. Then

$$w^{k(u,v)} \geq r_u^v, \quad u \geq 0, \quad v = 0, \dots, \eta(u) - 1.$$

Proof. We note that

$$\begin{aligned} w^{k(u,v)} &= \|a_u^{v+1}\|_{B^{k(u,v)}}^* \\ &\geq \|a_u^{v+1}\|_{H_u^v}^* \quad (\text{by Lemma 5.2}) \\ &= r_u^v. \end{aligned}$$

This proves the result. $\square$

LEMMA 5.4. *For any $k \geq 0$, we have*

$$\sum_{r=0}^k (w^r)^2 \leq 2m^2 \ln \left(1 + \frac{k+1}{8m^2} \right).$$

Proof. Using a result of Nesterov [19], Goffin, Luo, and Ye [14, Lemma 6.5] have achieved this upper bound. $\square$

We now consider Stopping Criterion 1 of the algorithm. This stopping rule implies that the total number of feasibility cuts produced by the algorithm does not exceed $k_{\epsilon, \delta}$, where $k_{\epsilon, \delta}$ is the smallest index, k , satisfying

$$(5.2) \quad \frac{1 + 4m \ln(1 + \frac{k}{8m^2})}{1 + 2m + k} \exp \left(-\frac{2c(\alpha, \gamma)k}{1 + 2m + k} - \frac{2\zeta(\alpha, \gamma)}{1 + 2m + k} \right) \leq \frac{(\epsilon\delta)^2}{2m}.$$

Hence, the total number of feasibility cuts produced by the algorithm is at most $O^*(m^2/(\delta\epsilon)^2)$, where the asterisk notation means that lower order additive and multiplicative terms are ignored. We will show that under Assumption 2, as soon as Stopping Criterion 1 is satisfied, the last feasible point found by the algorithm is an ϵ -solution of (CP). We first find an upper bound on the min-potential of the set of localization in the following lemma.

LEMMA 5.5. *Let $\epsilon > 0$ be a given tolerance and $0 < \delta < 1/2$ be as in Assumption 2. Then*

$$z(l) - f_* \geq \epsilon$$

implies that

$$P(\Omega_l^j) < (1 + 2m + k(l, j)) \ln \frac{2}{\delta\epsilon}$$

for all $l \geq 0$ and $j = 0, \dots, \eta(l)$.

Proof. Consider the ball $\mathcal{B}(\bar{y}, \rho(\Gamma_{\epsilon/2})) \subset \Gamma_{\epsilon/2}$. Then since $z(l) - f_* \geq \epsilon$ and $0 < \delta < 1/2$, we have

$$z(l) - f(\bar{y}) \geq f_* + \epsilon - f(\bar{y}) \geq \epsilon/2 > \epsilon\delta/2.$$

Now, in view of the definition of the min-potential, Assumption 2, and Corollary 2.10 we have

$$\begin{aligned}
P(\Omega_l^j) &= \min_{y \in \Omega_l^j} \left\{ -\ln(z(l) - f(y)) - \sum_{u=0}^{l-1} \sum_{v=1}^{\eta(u)} \ln(c_u^v - (a_u^v)^T y) \right. \\
&\quad \left. - \sum_{i=1}^j \ln(c_l^i - (a_l^i)^T y) - \sum_{i=1}^m \ln y_i - \sum_{i=1}^m \ln(1 - y_i) \right\} \\
&\leq -\ln(z(l) - f(\bar{y})) - \sum_{u=0}^{l-1} \sum_{v=1}^{\eta(u)} \ln(c_u^v - (a_u^v)^T \bar{y}) \\
&\quad - \sum_{i=1}^j \ln(c_l^i - (a_l^i)^T \bar{y}) - \sum_{i=1}^m \ln \bar{y}_i - \sum_{i=1}^m \ln(1 - \bar{y}_i) \\
&< \ln \frac{2}{\epsilon \delta} + (2m + k(l, j)) \ln \frac{1}{\rho(\Gamma_{\epsilon/2})} \\
&\leq (1 + 2m + k(l, j)) \ln \frac{2}{\epsilon \delta} \quad (\text{by Corollary 2.10 } \rho(\Gamma_{\epsilon/2}) \geq \epsilon \delta / 2).
\end{aligned}$$

This proves the result. $\square$

THEOREM 5.6. *Under Assumption 2, as soon as Stopping Criterion 1 is satisfied, the last feasible test point found by the algorithm is an ϵ -solution of (CP).*

Proof. Using Lemma 5.1, we can find the following lower bound on the min-potential:

$$(5.3) \quad P(\Omega_l^j) \geq P(\Omega^0) - \sum_{u=0}^{l-1} \sum_{v=0}^{\eta(l)-1} \ln r_u^v - \sum_{v=0}^{j-1} \ln r_l^v + c(\alpha, \gamma)k(l, j) + \zeta(\alpha, \gamma),$$

where $l \geq 0$ and $0 \leq j \leq \eta(l)$. It now follows from Lemma 5.5 and (5.3) that as long as $z(l) - f_* \geq \epsilon$, we have

$$\begin{aligned}
(1 + 2m + k(l, j)) \ln \frac{\epsilon \delta}{2} &< 2m \ln \frac{1}{2} + \sum_{u=0}^{l-1} \sum_{v=0}^{\eta(u)-1} \ln r_u^v + \sum_{v=0}^{j-1} \ln r_l^v \\
&\quad - c(\alpha, \gamma)k(l, j) - \zeta(\alpha, \gamma).
\end{aligned}$$

Now, by Lemma 5.3 and Lemma 5.4, we have

$$\begin{aligned}
(1 + 2m + k(l, j)) \ln \frac{\epsilon \delta}{2} &< 2m \ln \frac{1}{2} + \frac{1}{2} \sum_{i=0}^{k(l, j)-1} (w^i)^2 - c(\alpha, \gamma)k(l, j) - \zeta(\alpha, \gamma) \\
&\leq 2m \ln \frac{1}{2} + m^2 \ln \left(1 + \frac{k(l, j)}{8m^2} \right) - c(\alpha, \gamma)k(l, j) - \zeta(\alpha, \gamma),
\end{aligned}$$

from which, following the argument by Goffin, Luo, and Ye [14, Theorem 6.1], we conclude that

$$(5.4) \quad \frac{(\epsilon \delta)^2}{4m} < \frac{\frac{1}{2} + 2m \ln(1 + \frac{k(l, j)}{8m^2})}{1 + 2m + k(l, j)} \exp \left(-\frac{2c(\alpha, \gamma)k(l, j)}{1 + 2m + k(l, j)} - \frac{2\zeta(\alpha, \gamma)}{1 + 2m + k(l, j)} \right).$$

Thus, if inequality (5.4) is violated, we have $z(l) - f_* \leq \epsilon$, which implies that $f(y_{l-1}^f) - f_* < \epsilon$ since $z(l) > f(y_{l-1}^f)$. This means that as soon as Stopping Criterion 1 is satisfied by the algorithm, the last feasible point found is an ϵ -solution. $\square$

We will now consider Stopping Criterion 2 of the algorithm. The following lemma will lead to an upper bound on the gap $z(l) - f_*$, which is used in Stopping Criterion 2.

LEMMA 5.7. *Let y_l^a be the exact analytic center of Ω_l^f , where $l \geq 0$. Then*

$$z(l) - f_* \leq \{1 + 2m + k(l, \eta(l))\}(z(l) - f(y_l^a)).$$

Proof. Let $n(l) = 2m + k(l, \eta(l))$, and let, for simplicity, Ω_l^f be represented as follows:

$$\Omega_l^f = \{y \in R^m : f(y) \leq z(l), a_i^T y \leq c_i, i = 1, \dots, n(l)\}.$$

The exact analytic center y_l^a minimizes $\phi(\Omega_l^f, y)$. The necessary and sufficient conditions for this minimum are

$$(5.5) \quad \begin{cases} a_i^T y_l^a \leq c_i, & i = 1, \dots, n(l), \\ -\sum_{i=1}^{n(l)} x_i a_i = \nabla f(y_l^a), & x \geq 0, \\ x_i(c_i - a_i^T y_l^a) = z(l) - f(y_l^a), & i = 1, \dots, n(l). \end{cases}$$

Let $\Gamma_l^f = \{y \in R^m : a_i^T(y) \leq c_i, i = 1, \dots, n(l)\}$. Then, by Wolfe's formulation of duality [24], the dual of $\min\{f(y) : y \in \Gamma_l^f\}$ is given by

$$\max_{x, y} \left\{ f(y) + \sum_{i=1}^{n(l)} x_i(a_i^T y - c_i) : -\sum_{i=1}^{n(l)} x_i a_i = \nabla f(y), x \geq 0 \right\}.$$

Since (x, y_l^a) is a feasible solution for this dual, we have

$$f_*^l \geq f(y_l^a) + \sum_{i=1}^{n(l)} x_i(a_i^T y_l^a - c_i),$$

where f_*^l is the optimal value of f on Γ_l^f . But $f_* \geq f_*^l$ since $\Gamma \subset \Gamma_l^f$. Therefore,

$$\begin{aligned} f(y_l^a) - f_* &\leq \sum_{i=1}^{n(l)} x_i(c_i - a_i^T y_l^a) \\ &= n(l)(z(l) - f(y_l^a)). \end{aligned}$$

This implies that

$$\begin{aligned} z(l) - f_* &\leq z(l) - f(y_l^a) + f(y_l^a) - f_* \\ &\leq (1 + n(l))(z(l) - f(y_l^a)). \end{aligned}$$

This proves the result. $\square$

THEOREM 5.8. *If Stopping Criterion 2 of the algorithm is satisfied at a feasible test point, then this point is an ϵ -solution of the problem.*

Proof. Let $n(l) = 2m + k(l, \eta(l))$. Then Lemma 5.7, together with Lemma 4.1 ((4.1)) and the fact that in the algorithm $0 < \gamma \leq (\sqrt{2} - 1)^2/2$, imply that if y_l^f , where $l \geq 0$, satisfies Stopping Criterion 2, then

$$\begin{aligned} z(l) - f_* &\leq (1 + n(l))(z(l) - f(y_l^a)) \\ &\leq \left(1 + \frac{\gamma}{(2 - \sqrt{2})\sqrt{\alpha}}\right) (1 + n(l))(z(l) - f(y_l^f)) \\ &\leq \frac{5}{4}(1 + n(l))(z(l) - f(y_l^f)) \\ &\leq \epsilon. \end{aligned}$$

This means that $f(y_l^f) - f_* < \epsilon$; thus, y_l^f is an ϵ -solution of problem (CP). $\square$

Now let

$$(5.6) \quad \hat{\epsilon} = \frac{4\epsilon(1 - \theta)}{5(1 + 2m + k_{\epsilon, \delta})}.$$

Then we have the following result.

LEMMA 5.9. *If, for $l \geq 1$,*

$$z(l - 1) \geq f(y_{l-1}^f) + \frac{4\epsilon}{5(1 + 2m + k(l - 1, \eta(l - 1)))},$$

then for all $j = 0, \dots, \eta(l)$, we have

$$P(\Omega_l^j) \leq (1 + 2m + k(l, j)) \ln \frac{\beta^*}{\hat{\epsilon}},$$

where

$$\beta^* = 2 \max \left\{ 1, \frac{z(0) - f_*}{\rho(\Gamma_{z(0) - f_*})} \right\}.$$

Proof. Note that

$$\begin{aligned} z(l) - f_* &\geq z(l) - f(y_{l-1}^f) \\ &= (1 - \theta)(z(l - 1) - f(y_{l-1}^f)) \\ &\geq (1 - \theta) \frac{4\epsilon}{5(1 + 2m + k(l - 1, \eta(l - 1)))} \\ &\geq \hat{\epsilon}. \end{aligned}$$

Now consider the ball $\mathcal{B}(\hat{y}, \rho(\Gamma_{\hat{\epsilon}/2})) \subset \Gamma_{\hat{\epsilon}/2}$. Then

$$z(l) - f(\hat{y}) \geq f_* + \hat{\epsilon} - f(\hat{y}) \geq \hat{\epsilon}/2.$$

Also, in view of Corollary 2.9, we have

$$\rho(\Gamma_{\hat{\epsilon}/2}) \geq \frac{\hat{\epsilon}\rho(z(0) - f_*)}{2(z(0) - f_*)}.$$

Therefore, we have

$$\begin{aligned}
P(\Omega_l^j) &= \min_{y \in \Omega_l^j} \left\{ -\ln(z(l) - f(y)) - \sum_{u=0}^{l-1} \sum_{v=1}^{\eta(u)} \ln(c_u^v - (a_u^v)^T y) \right. \\
&\quad \left. - \sum_{i=1}^j \ln(c_l^i - (a_l^i)^T y) - \sum_{i=1}^m \ln y_i - \sum_{i=1}^m \ln(1 - y_i) \right\} \\
&\leq -\ln(z(l) - f(\hat{y})) - \sum_{u=0}^{l-1} \sum_{v=1}^{\eta(u)} \ln(c_u^v(l, j, t) - (a_u^v)^T \hat{y}) \\
&\quad - \sum_{i=1}^j \ln(c_l^i - (a_l^i)^T \hat{y}) - \sum_{i=1}^m \ln \hat{y}_i - \sum_{i=1}^m \ln(1 - \hat{y}_i) \\
&\leq \ln \frac{2}{\hat{\epsilon}} + (2m + k(l, j)) \ln \frac{1}{\rho(\Gamma_{\hat{\epsilon}/2})} \\
&\leq \ln \frac{2}{\hat{\epsilon}} + (2m + k(l, j)) \ln \frac{2(z(0) - f_*)}{\hat{\epsilon} \rho(\Gamma_{z(0)-f_*})} \\
&\leq (1 + 2m + k(l, j)) \ln \frac{\beta^*}{\hat{\epsilon}}.
\end{aligned}$$

This proves the result. $\square$

We can now derive an upper bound on the total number of iterations performed by the algorithm.

THEOREM 5.10. *The total number of iterations of the algorithm to find an ϵ -solution is*

$$O\left(\frac{(m^2 + k_{\epsilon, \delta})}{\theta} \ln \frac{(m + k_{\epsilon, \delta})\beta^*}{(1 - \theta)\epsilon}\right),$$

where β^* is as given in Lemma 5.9 and $k_{\epsilon, \delta}$ is the smallest index satisfying (5.2) and the constant factor in $O(\cdot)$ depends on α and on the choice of γ only.

Proof. As long as,

$$z(l-1) \geq f(y_{l-1}^f) + \frac{4\epsilon}{5(1 + 2m + k(l-1, \eta(l-1)))},$$

where $l \geq 1$, by Lemma 5.1 and Lemma 5.9, we have

$$\begin{aligned}
(1 + 2m + k(l, j)) \ln \frac{\beta^*}{\hat{\epsilon}} &\geq -2m \ln \frac{1}{2} - \sum_{u=0}^{l-1} \sum_{v=0}^{\eta(l)-1} \ln r_u^v - \sum_{v=0}^{j-1} \ln r_l^v + \lambda(\alpha, \gamma)\theta l \\
&\quad + c(\alpha, \gamma)k(l, j) + \zeta(\alpha, \gamma).
\end{aligned}$$

Hence, in view of Lemma 5.4, we have

$$\begin{aligned}
\lambda(\alpha, \gamma)\theta l &\leq (1 + 2m + k(l, j)) \ln \frac{\beta^*}{\hat{\epsilon}} + 2m \ln \frac{1}{2} - c(\alpha, \gamma)(k(l, j) - \zeta(\alpha, \gamma)) \\
&\quad + \sum_{u=0}^{l-1} \sum_{v=0}^{\eta(l)-1} \ln r_u^v + \sum_{v=0}^{j-1} \ln r_l^v \\
&\leq (1 + 2m + k(l, j)) \ln \frac{\beta^*}{\hat{\epsilon}} + 2m \ln \frac{1}{2} - c(\alpha, \gamma)k(l, j) - \zeta(\alpha, \gamma)
\end{aligned}$$

$$\begin{aligned}
& + \frac{1}{2} \sum_{r=0}^{k(l,j)-1} \ln(w^r)^2 \\
& \leq (1 + 2m + k(l, j)) \ln \frac{\beta^*}{\hat{\epsilon}} + 2m \ln \frac{1}{2} - c(\alpha, \gamma)k(l, j) - \zeta(\alpha, \gamma) \\
& \quad + m^2 \ln \left(1 + \frac{k(l, j)}{8m^2} \right).
\end{aligned}$$

Now, since the total number of feasibility cuts is bounded by $k_{\epsilon, \delta}$, we have

$$\begin{aligned}
(5.7) \quad l & \leq \frac{1}{\lambda(\alpha, \gamma)\theta} \left[(1 + 2m + k(l, j)) \ln \frac{\beta^*}{\hat{\epsilon}} + 2m \ln \frac{1}{2} \right. \\
& \quad \left. - c(\alpha, \gamma)k(l, j) - \zeta(\alpha, \gamma) + m^2 \ln \left(1 + \frac{k(l, j)}{8m^2} \right) \right] \\
& \leq \frac{1}{\lambda(\alpha, \gamma)\theta} \left[(1 + 2m + k_{\epsilon, \delta}) \ln \frac{\beta^*}{\hat{\epsilon}} + 2m \ln \frac{1}{2} \right. \\
& \quad \left. + |c(\alpha, \gamma)|k_{\epsilon, \delta} - \zeta(\alpha, \gamma) + m^2 \ln \left(1 + \frac{k_{\epsilon, \delta}}{8m^2} \right) \right].
\end{aligned}$$

Note that the left-hand side of inequality (5.7) represents the total number of translations of the objective constraint performed during the algorithm. Let $T_{\epsilon, \delta}$ be the smallest index that is greater than the right-hand side of inequality (5.7). Suppose that Stopping Criterion 1 is not satisfied. As soon as inequality (5.7) is violated, we have

$$z(l-1) - f(y_{l-1}^f) \leq \frac{4\epsilon}{5(1 + 2m + k(l-1, \eta(l-1)))};$$

i.e., Stopping Criterion 2 is satisfied. Hence the total number of translations of the objective constraint does not exceed $T_{\epsilon, \delta}$. Since the total number of feasibility cuts is $k_{\epsilon, \delta}$, in no more than $T_{\epsilon, \delta} + k_{\epsilon, \delta}$ iterations, Stopping Criterion 2 will be satisfied. If Stopping Criterion 1 is satisfied, then inequality (5.7) holds and again the total number of iterations will be bounded by $T_{\epsilon, \delta} + k_{\epsilon, \delta}$. Since this upper bound is $O(\frac{(m^2 + k_{\epsilon, \delta})}{\theta} \ln \frac{\beta^*}{\hat{\epsilon}})$, the result follows. $\square$

6. Update to a new center. In each iteration of the algorithm, we need to calculate a γ -approximate analytic center (with $0 < \gamma \leq (\sqrt{2} - 1)^2/2$) of a set of localization. In this section, we will study how to calculate a new approximate analytic center in the following cases:

1. when a feasibility cut is added;
2. when the nonlinear objective cut is added; and
3. when the objective cut is shifted.

We will also give upper bounds on the total number of (damped) Newton steps required to recenter. Throughout this section, we assume that $\bar{N}$ is defined by (2.6).

THEOREM 6.1. *In the algorithm, the total number of (damped) Newton steps to update to a new γ -approximate analytic center after addition of a linear cut is at most $O(1)$, with $O(1)$ depending on α and γ only.*

Proof. We choose y^+ according to (4.4) of Lemma 4.4 and use the same notation as in that lemma. Hence, we have

$$(6.1) \quad \phi(\Omega^+, y^+) - P(\Omega^+) \leq \bar{c}(\alpha, \gamma) - c(\alpha, \gamma),$$

where $c(\alpha, \gamma)$ and $\bar{c}(\alpha, \gamma)$ are given by (4.6) and (4.7). Now let $q(\Omega^+, y)$ be the Newton direction of $\phi(\Omega^+, y)$ at y . By Lemma 2.4, as long as

$$\sqrt{\alpha} \|g(\Omega^+, y)\|_{H(\Omega^+, y)}^* \geq \gamma_*,$$

there is the following minimal decrease in the min-potential at each iteration:

$$(6.2) \quad \phi(\Omega^+, y) - \phi(\Omega^+, y + \sigma q(\Omega^+, y)) \geq \omega(\alpha),$$

where

$$(6.3) \quad \omega(\alpha) = \{\gamma_* - \ln(1 + \gamma_*)\}/\alpha$$

with γ_* given by (2.4). It now follows from (6.1) and (6.2) that, starting from y^+ , in no more than

$$\left\lceil \frac{\bar{c}(\alpha, \gamma) - c(\alpha, \gamma)}{\omega(\alpha)} \right\rceil$$

damped Newton steps, a γ_* -approximate analytic center of Ω^+ is calculated. Furthermore, by Lemma 2.5, at most another $\bar{N}$ (depending on γ only) full Newton steps are needed to calculate a γ -approximate analytic center of Ω^+ . $\square$

At the beginning of the algorithm, the cut corresponding to the objective, which is a nonlinear cut, is added. The next theorem shows how to update to a new center in this case.

THEOREM 6.2. *Consider $w^0 \in \Gamma_{int}$, which is the analytic center of Ω^0 . It takes $O(1)$ (damped) Newton steps to calculate a γ -approximate analytic center of Ω_0^0 . Here, $O(1)$ depends on α and γ only.*

Proof. By Lemma 4.3 ((4.3)), we have

$$\|g(\Omega_0^0, w^0)\|_{H(\Omega_0^0, w^0)}^* \leq \|g(\Omega^0, w^0)\|_{H(\Omega^0, w^0)}^* + \frac{\gamma}{\sqrt{\alpha}}.$$

Recall from the algorithm that $0 < \gamma \leq (\sqrt{2} - 1)^2/2$. Hence

$$\|g(\Omega_0^0, w^0)\|_{H(\Omega_0^0, w^0)}^* \leq \frac{2\gamma}{\sqrt{\alpha}} < \frac{1}{3\sqrt{\alpha}}.$$

It now follows from Lemma 2.3 that

$$\phi(\Omega_0^0, w^0) - P(\Omega_0^0) \leq d(\alpha, \gamma),$$

where

$$(6.4) \quad d(\alpha, \gamma) = \frac{4\gamma^2}{\alpha\{1 - (\frac{9}{2}\gamma)^2\}}.$$

Now, using the damped Newton method, starting at w^0 , it takes no more than

$$\left\lceil \frac{d(\alpha, \gamma)}{\omega(\alpha)} \right\rceil$$

damped Newton steps (with $\omega(\alpha)$ defined by (6.3)) to find a γ_* -approximate analytic center of Ω_0^0 . Finally, by Lemma 2.5, it takes another $\bar{N}$ Newton steps to calculate a γ -approximate analytic center of Ω_0^0 . $\square$

Finally, we consider the case where the objective constraint is shifted. The next lemma is due to Den Hertog [5, Theorem 3.4], which is also a generalization of an inequality of Vaidya [23] for the linear programming (LP) case to the convex case, adapted to our case.

LEMMA 6.3. *Consider the bounded set*

$$(6.5) \quad \Omega(\nu) = \{y \in R^m : a_i^T y \leq c_i, i = 1, \dots, n, f(y) \leq \nu\}$$

with a nonempty interior. Then, as ν decreases, $\{\nu - f(y_\nu^a)\}$ decreases monotonically. Here y_ν^a is the exact analytic center of $\Omega(\nu)$.

THEOREM 6.4. *Consider the set Ω as defined by (2.1). Let*

$$\Omega_\theta = \{y \in R^m : a_i^T y \leq c_i, i = 1, \dots, n, f(y) \leq z_\theta\},$$

where $z_\theta = z - \theta(z - f(y^c))$, $0 < \theta < 1$ and y^c is a γ -approximate analytic center of Ω . Then the total number of (damped) Newton iterations to find a γ -approximate analytic center of Ω_θ is at most $O(1)$, where $O(1)$ depends on α , γ , and θ only.

Proof. By Lemma 2.3, we have

$$(6.6) \quad \phi(\Omega, y^c) - P(\Omega) \leq \chi(\alpha, \gamma),$$

where

$$(6.7) \quad \chi(\alpha, \gamma) = \frac{\gamma^2}{\alpha\{1 - (\frac{9}{4}\gamma)^2\}}.$$

Let $\Omega(\nu)$ be as defined by (6.5), and let y^a and y_ν^a be the exact analytic centers of Ω and $\Omega(\nu)$, respectively. Consider

$$\tilde{\phi}(y, \nu) = \phi(\Omega(\nu), y) - P(\Omega(\nu)).$$

We note that

$$\begin{aligned} \tilde{\phi}(y, z) &= \phi(\Omega, y) - P(\Omega), \\ \tilde{\phi}(y, z_\theta) &= \phi(\Omega_\theta, y) - P(\Omega_\theta). \end{aligned}$$

We will use an argument due to Den Hertog [5, Theorem 3.6] to find an upper bound on $\tilde{\phi}(y, z_\theta)$. By mean value theorem, there exists $\hat{\nu} \in (z_\theta, z)$ such that for all $y \in \Omega_\theta$,

$$\tilde{\phi}(y, z_\theta) = \tilde{\phi}(y, z) - \left. \frac{d}{d\nu} \tilde{\phi}(y, \nu) \right|_{\nu=\hat{\nu}} (z - z_\theta).$$

But

$$\frac{d}{d\nu} \tilde{\phi}(y, \nu) = \frac{-1}{\nu - f(y)} - \frac{d}{d\nu} P(\Omega(\nu))$$

and

$$\begin{aligned} \frac{d}{d\nu} P(\Omega(\nu)) &= \sum_{i=1}^n \frac{a_i \frac{dy_\nu^a}{d\nu}}{c_i - a_i^T y_\nu^a} + \frac{\nabla f(y_\nu^a) \frac{dy_\nu^a}{d\nu} - 1}{\nu - f(y_\nu^a)} \\ &= \frac{-1}{\nu - f(y_\nu^a)} \quad (\text{since } y_\nu^a \text{ minimizes } \phi(\Omega(\nu), y)). \end{aligned}$$

Therefore,

$$\frac{d}{d\nu}\tilde{\phi}(y, \nu) = \frac{-1}{\nu - f(y)} + \frac{1}{\nu - f(y_\nu^a)},$$

which, in view of Lemma 6.3, implies that

$$\begin{aligned}\tilde{\phi}(y, z_\theta) &= \tilde{\phi}(y, z) - \left\{ -\frac{1}{\nu - f(y)} + \frac{1}{\nu - f(y_\nu^a)} \right\} \Big|_{\nu=\hat{\nu}} (z - z_\theta) \\ &\leq \tilde{\phi}(y, z) - \left\{ -\frac{1}{z_\theta - f(y)} + \frac{1}{z - f(y^a)} \right\} (z - z_\theta).\end{aligned}$$

Now, by (6.6) and the definition of z_θ , we have

$$\begin{aligned}\tilde{\phi}(y^c, z_\theta) &\leq \chi(\alpha, \gamma) - \left\{ \frac{-\theta}{1-\theta} + \theta \frac{z - f(y^c)}{z - f(y^a)} \right\} \\ &\leq \chi(\alpha, \gamma) + \theta \left\{ \frac{1}{1-\theta} - \frac{1}{1 + \frac{\gamma}{(2-\sqrt{2})\sqrt{\alpha}}} \right\} \quad (\text{by Lemma 4.1}).\end{aligned}$$

Thus,

$$(6.8) \quad \phi(\Omega_\theta, y^c) - P(\Omega_\theta) \leq \chi(\alpha, \gamma) + \theta \left\{ \frac{1}{1-\theta} - \frac{1}{1 + \frac{\gamma}{(2-\sqrt{2})\sqrt{\alpha}}} \right\}.$$

Hence, in no more than

$$\left\lceil \frac{\chi(\alpha, \gamma) + \theta \left\{ \frac{1}{1-\theta} - \frac{1}{1 + \frac{\gamma}{(2-\sqrt{2})\sqrt{\alpha}}} \right\}}{\omega(\alpha)} \right\rceil$$

damped Newton iterations ($\omega(\alpha)$ as in (6.3)), a γ_* -approximate analytic center of Ω_θ is calculated. Since $\gamma < \gamma_*$, we can use full Newton steps according to Lemma 2.5 to calculate a γ -approximate analytic center of Ω_θ in at most another $\bar{N}$ full Newton steps. $\square$

COROLLARY 6.5. *In the algorithm, the total number of (damped) Newton steps to update to a new γ -approximate analytic center after a translation of the objective constraint is $O(1)$, with $O(1)$ depending on α , γ , and θ only.*

We can now conclude the following result.

THEOREM 6.6. *The cutting plane algorithm finds an ϵ -solution of (CP) in at most*

$$O\left(\frac{(m^2 + k_{\epsilon, \delta})}{\theta} \ln \frac{(m + k_{\epsilon, \delta})\beta^*}{(1-\theta)\epsilon}\right)$$

(damped) Newton iterations, where β^* is as given in Lemma 5.9 and $k_{\epsilon, \delta}$ is the smallest index satisfying (5.2) and the constant factor in $O(\cdot)$ depends on α and on the choice of γ only.

The overall complexity in terms of the O^* notation is $O^*\left(\frac{m^2}{(\epsilon\delta)^2}\right)$.

7. Additional remarks. In our algorithm, we did not discuss the issue of parallel cuts. If the oracle generates a feasibility cut parallel to an existing one, the old cut becomes redundant, and one may wish to discard it. Equivalently, the old cut is shifted. There are many general cases of (CP) , e.g., when Γ is defined by (nonsmooth) convex constraints, in which the issue of parallel cuts is not important, and the chance of the oracle producing parallel cuts is little. Furthermore, in these cases, even if the oracle produces parallel cuts, the algorithm may fail to detect it due to round-off errors. On the other hand, there are some special cases in which the issue of parallel cuts becomes important and useful, for example, the case where Γ is defined by a finite (but large) number of linear constraints. Problems involving a nonlinear objective and a finite (yet large) number of linear constraints appear a lot in practice, e.g., in multicommodity flows or water resources problems. In the special case of having a finite number of linear constraints, the oracle simply looks for a violated constraint and produces a cut with the same normal vector as the violated constraint. In this case, therefore, the chances of producing several parallel cuts are quite high. Although not considering translations does not endanger the convergence of the algorithm applied to this class of problems, it is more useful to consider translation of feasibility cuts. This is because, if we assume the number of constraints is N , then by including translations of feasibility cuts, we guarantee that the number of feasibility cuts produced by the algorithm remains bounded by N . The other advantage is that in this case we can drop Stopping Criterion 1.

Hence, we can generalize the algorithm so that it includes translations of feasibility cuts all the way to the approximate analytic center of the current set of localization as well. The complexity analysis of this generalized algorithm can be handled in a very similar way at the cost of some technicalities. In fact, the same complexity bound can be obtained for the generalized algorithm. We refer the reader interested in this general case to our technical report [21]. Furthermore, as is discussed in [21], for the special case where the constraints are linear (and there is N of them), we can drop Stopping Criterion 1 in the generalized algorithm to obtain an algorithm which has a polynomial worst case complexity in N , i.e., $O(N \ln(N/\epsilon))$.

8. Appendix. In this appendix, we give a proof of Lemma 4.4 which is basically an extension of Lemma 2.1 of Nesterov [19] adapted to our Ω as defined by (2.1). The proof is very similar, yet some constants change due to the fact that our $\phi(\Omega, y)$ is strongly α -self-concordant. We need to state two lemmas that will be used in the proof. The following lemma is taken from Nesterov and Nemirovskii [20, Theorem 2.1.1].

LEMMA 8.1. *Let Ω be defined by (2.1) and $x \in \Omega_{int}$. Then for every y such that*

$$(8.1) \quad r = \sqrt{\alpha} \|x - y\|_{H(\Omega, x)} < 1$$

we have

$$(1 - r)^2 H(\Omega, x) \leq H(\Omega, y) \leq \frac{1}{(1 - r)^2} H(\Omega, x).$$

Furthermore, $y \in \Omega_{int}$.

The next lemma is a trivial extension of Theorem 2.3 of [19] adapted to our Ω . The result there is for strongly 1-self-concordant functions. We can simply extend it to strongly α -self-concordant functions by noticing that if $\phi(\Omega, y)$ is α -self-concordant, then $\alpha\phi(\Omega, y)$ is 1-self-concordant.

LEMMA 8.2. *Under the same assumptions as in Lemma 8.1, we have the following:*

$$1. \quad \alpha\phi(\Omega, y) \leq \alpha\phi(\Omega, x) + \langle \alpha g(\Omega, x), y - x \rangle - (\ln(1 - r) + r).$$

2. $\alpha\phi(\Omega, x) \leq \alpha\phi(\Omega, y) + \langle \alpha g(\Omega, y), x - y \rangle + \ln(1 - r) + r/(1 - r)$.
 3. $\sqrt{\alpha}\|g(\Omega, y) - g(\Omega, x) - H(\Omega, x)(y - x)\|_{H(\Omega, x)}^* \leq r(1/(1 - r) - 1)$.
- Proof of Lemma 4.4.* Consider a central path

$$y(t) = \arg \min\{\phi(\Omega, u) : \langle a_{n+1}, y^c - u \rangle = t\}, \quad 0 \leq t \leq t^*,$$

where

$$t^* = \sup\{\langle a_{n+1}, y^c - u \rangle : u \in \Omega\}.$$

As is argued in [19], this path is well defined. By the first order optimality conditions, for any $t \in [0, t^*)$, there exists $\lambda(t) \geq 0$ such that

$$(8.2) \quad g(\Omega, y(t)) + \lambda(t)a_{n+1} = 0.$$

Let y_+^a be the exact analytic center of Ω^+ . Then for any $t \in [0, t^*)$, we have

$$\begin{aligned} P(\Omega^+) &\geq \phi(\Omega, y(t)) + \langle g(\Omega, y(t)), y_+^a - y(t) \rangle - \ln \tau \\ &= \phi(\Omega, y(t)) + \langle g(\Omega, y(t)), y^c - y(t) \rangle + \lambda(t)\tau - \ln(\tau) \end{aligned}$$

with $\tau = \langle a_{n+1}, y^c - y_+^a \rangle > 0$. Using the inequality

$$\lambda\tau - \ln \tau \geq 1 + \ln \lambda,$$

valid for all positive λ and τ , we get

$$(8.3) \quad P(\Omega^+) \geq \phi(\Omega, y(t)) + \langle g(\Omega, y(t)), y^c - y(t) \rangle + 1 + \ln \lambda(t).$$

Since y^c is a γ -approximate analytic center of Ω , it follows from Lemma 2.2 that

$$\sqrt{\alpha}\|y(0) - y^c\|_{H(\Omega, y^c)} \leq 1 - \frac{1}{\sqrt{2}} \equiv r.$$

We can choose t in such a way that this inequality becomes an equality:

$$\sqrt{\alpha}\|y(0) - y^c\|_{H(\Omega, y^c)} = r.$$

We now find a lower bound on $\lambda(t)$. By Lemma 8.2 (3) and definition of r , we have

$$\begin{aligned} \alpha\lambda(t)t &= \langle \alpha g(\Omega, y(t)), y(t) - y^c \rangle \\ &= \langle \alpha g(\Omega, y(t)) - \alpha g(\Omega, y^c) - \alpha H(\Omega, y^c)(y(t) - y^c), y(t) - y^c \rangle \\ &\quad + \alpha \langle g(\Omega, y^c), y(t) - y^c \rangle + r^2 \\ &\geq -r\sqrt{\alpha}\|g(\Omega, y(t)) - g(\Omega, y^c) - H(\Omega, y^c)(y(t) - y^c)\|_{H(\Omega, y^c)}^* \\ &\quad - \sqrt{\alpha}\|g(\Omega, y^c)\|_{H(\Omega, y^c)}^* r + r^2 \\ &\geq -r^2 \left(\frac{1}{1-r} - 1 \right) - r\gamma + r^2 \\ (8.4) \quad &= r((\sqrt{2} - 1)^2 - \gamma), \end{aligned}$$

$$\begin{aligned} \alpha\lambda(t)(\|a_{n+1}\|_{\alpha H(\Omega, y^c)}^*)^2 &= \lambda(t)(\|a_{n+1}\|_{H(\Omega, y^c)}^*)^2 \\ &= -\langle H^{-1}(\Omega, y^c)a_{n+1}, g(\Omega, y(t)) \rangle \\ &\geq t - \langle H^{-1}(\Omega, y^c)a_{n+1}, g(\Omega, y(t)) \rangle \\ &\quad + \langle H^{-1}(\Omega, y^c)a_{n+1}, g(\Omega, y^c) + H(\Omega, y^c)(y(t) - y^c) - g(\Omega, y^c) \rangle \\ (8.5) \quad &\geq t - \left(\gamma + r \left(\frac{1}{1-r} - 1 \right) \right) \|a_{n+1}\|_{\alpha H(\Omega, y^c)}^*. \end{aligned}$$

Substituting inequality (8.5) into (8.4) by eliminating t , we get

$$\alpha\lambda(t)\|a_{n+1}\|_{\alpha H(\Omega, y^c)}^* \left(\alpha\lambda(t)\|a_{n+1}\|_{\alpha H(\Omega, y^c)}^* + \gamma + r \left(\frac{1}{1-r} - 1 \right) \right) \geq r((\sqrt{2}-1)^2 - \gamma).$$

This quadratic inequality gives the following lower bound for $\lambda(t)$:

$$(8.6) \quad \lambda(t) \geq \frac{(\sqrt{2}-1)^2 - \gamma}{\sqrt{\alpha}\|a_{n+1}\|_{H(\Omega, y^c)}^*}.$$

(One can easily check that the right-hand side of (8.6) satisfies the quadratic inequality as an equality.) Now, by (8.3), Lemma 8.2 (2), and (8.6), we have

$$\begin{aligned} P(\Omega^+) &\geq \phi(\Omega, y^c) - \frac{1}{\alpha} \left(\ln(1-r) + \frac{r}{1-r} \right) + 1 + \ln \lambda(t) \\ &\geq \phi(\Omega, y^c) - \ln \|a_{n+1}\|_{H(\Omega, y^c)}^* + c(\alpha, \gamma), \end{aligned}$$

where

$$c(\alpha, \gamma) = 1 - (\sqrt{2} - \ln \sqrt{2} - 1)/\alpha + \ln[((\sqrt{2}-1)^2 - \gamma)/\sqrt{\alpha}].$$

To get the upper bound for $P(\Omega^+)$, we use Lemma 8.2 (1) as follows:

$$\begin{aligned} \phi(\Omega^+, y^+) &= \phi(\Omega, y^+) - \ln \langle a_{n+1}, y^c - y^+ \rangle \\ &= \phi(\Omega, y^+) - \ln \|a_{n+1}\|_{H(\Omega, y^c)}^* + \ln(3\sqrt{\alpha}) \\ &\leq \phi(\Omega, y^c) + \langle g(\Omega, y^c), y^+ - y^c \rangle - (\ln(2/3) + 1/3)/\alpha \\ &\quad - \ln \|a_{n+1}\|_{H(\Omega, y^c)}^* + \ln(3\sqrt{\alpha}) \\ &\leq \phi(\Omega, y^c) + \gamma/(3\alpha) - (\ln(2/3) + 1/3)/\alpha - \ln \|a_{n+1}\|_{H(\Omega, y^c)}^* + \ln(3\sqrt{\alpha}) \\ &= \phi(\Omega, y^c) - \ln \|a_{n+1}\|_{H(\Omega, y^c)}^* + \bar{c}(\alpha, \gamma), \end{aligned}$$

where

$$\bar{c}(\alpha, \gamma) = \gamma/(3\alpha) + \ln(3\sqrt{\alpha}) - (\ln(2/3) + 1/3)/\alpha. \quad \square$$

Acknowledgment. We would like to thank the referees for many helpful suggestions on improving the presentation of the paper.

REFERENCES

- [1] A. ALTMAN AND K. C. KIWIEL, *A note on some analytic center cutting plane methods for convex feasibility and minimization problems*, Comput. Optim. Appl., 5 (1996), pp. 175–180.
- [2] D. S. ATKINSON AND P. M. VAIDYA, *A cutting plane algorithm for convex programming that uses analytic centers*, Math. Programming, 69 (1995), pp. 1–43.
- [3] O. BAHN, O. DU MERLE, J. L. GOFFIN, AND J. P. VIAL, *A cutting plane method from analytic centers for stochastic programming*, Math. Programming, 69 (1995), pp. 45–73.
- [4] O. BAHN, J. L. GOFFIN, J. P. VIAL, AND O. DU MERLE, *Experimental behavior of an interior point cutting plane algorithm for convex programming: An application to geometric programming*, Discrete Appl. Math., 49 (1994), pp. 3–23.
- [5] D. DEN HERTOG, *Interior-Point Approach to Linear, Quadratic and Convex Programming, Algorithms and Complexity*, Kluwer Publishers, Dordrecht, The Netherlands, 1994.
- [6] D. DEN HERTOG, J. KALISKI, C. ROOS, AND T. TERLAKY, *A logarithmic barrier cutting plane method for convex programming*, Ann. Oper. Res., 58 (1995), pp. 69–98.

- [7] J. KALISKI, D. HAGLIN, C. ROOS, AND T. TERLAKY, *Logarithmic barrier decomposition methods for semi-infinite programming*, Research Report, No. 96-51, Faculty of Technical Mathematics and Informatics, Delft University of Technology, The Netherlands, 1996.
- [8] H. G. EGGLESTON, *Convexity*, Cambridge University Press, London, 1969.
- [9] J. L. GOFFIN, *Convergence rates of the ellipsoid method on general convex functions*, Math. Oper. Res., 8 (1983), pp. 135–150.
- [10] J. L. GOFFIN, J. GONDZIO, R. SARKISSIAN, AND J. P. VIAL, *Solving nonlinear multicommodity problems by the analytic center cutting plane method*, Math. Programming, 75 (1997), pp. 131–154.
- [11] J. L. GOFFIN, A. HAURIE, AND J. P. VIAL, *Decomposition and nondifferentiable optimization with the projective algorithm*, Management Science, 37 (1992), pp. 284–302.
- [12] J. L. GOFFIN, A. HAURIE, J. P. VIAL, AND D. L. ZHU, *Using central prices in the decomposition of linear programs*, European Journal of Operations Research, 64 (1993), pp. 393–409.
- [13] J. L. GOFFIN, Z. Q. LUO, AND Y. YE, *On the complexity of a algorithm for convex or quasiconvex problems*, in Large Scale Optimization: State of the Art, W. W. Hager, D. W. Hearn, and P. M. Pardalos, eds., Kluwer Academic Publishers, Norwell, MA, 1993, pp. 187–196.
- [14] J. L. GOFFIN, Z. Q. LUO, AND Y. YE, *Complexity analysis of an interior point cutting plane method for convex feasibility problems*, SIAM J. Optim., 6 (1996), pp. 638–652.
- [15] J. L. GOFFIN AND J. P. VIAL, *Shallow, deep and very deep cuts in the analytic center cutting plane method*, Math. Programming, to appear.
- [16] K. C. KIWIEL, *Efficiency of the analytic center cutting plane method for convex minimization*, SIAM J. Optim., 7 (1997), pp. 336–346.
- [17] Z. Q. LUO AND J. SUN, *An analytic center based column generation algorithm for convex quadratic feasibility problems*, preprint, Department of Electrical and Computer Engineering, McMaster University, Hamilton, Ontario, Canada, 1995.
- [18] Z. Q. LUO AND J. SUN, *Cutting surfaces and analytic center: A polynomial algorithm for a convex feasibility problem defined by self-concordant inequalities*, preprint, Department of Electrical and Computer Engineering, McMaster University, Hamilton, Ontario, Canada, 1996.
- [19] Y. E. NESTEROV, *Complexity estimates of some cutting plane methods based on the analytical barrier*, Math. Programming, 69 (1995), pp. 149–176.
- [20] Y. E. NESTEROV AND A. S. NEMIROVSKII, *Interior-Point Polynomial Algorithms in Convex Programming*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1994.
- [21] F. SHARIFI MOKHTARIAN AND J. L. GOFFIN, *A Nonlinear Analytic Center Cutting Plane Method for a Class of Convex Programming Problems*, Technical Report G-96-53, Groupe d'études et de recherche en analyse des décisions, Université de Montréal, Canada, 1996.
- [22] G. SONNEVEND, *New algorithms in convex programming based on a notion of "center" (for systems of analytic inequalities) and on rational extrapolation* in Trends in Mathematical Optimization: Proceedings of the 4th French-German Conference on Optimization in Irsee, West Germany, April 1986, K. H. Hoffman, J. B. Hiriart-Urruty, C. Lemaréchal, and J. Zowe, eds., Internat. Ser. Numer. Math. 84, Birkhäuser-Verlag, Basel, Switzerland, 1988, pp. 311–327.
- [23] P. M. VAIDYA, *An algorithm for linear programming which requires $O(((m+n)n^2 + (m+n)^{1.5}n)L)$ arithmetic operations*, Math. Programming, 47 (1990), pp. 175–201.
- [24] PH. WOLFE, *A duality theorem for nonlinear programming*, Quart. Appl. Math., 19 (1961), pp. 239–244.
- [25] Y. YE, *A potential reduction algorithm allowing column generation*, SIAM J. Optim., 2 (1992), pp. 7–20.
- [26] Y. YE, *Complexity analysis of the analytic center cutting plane method that uses multiple cuts*, Technical Report, Department of Management Sciences, University of Iowa, Iowa City, IA, 1994.

PRIMAL-DUAL INTERIOR METHODS FOR NONCONVEX NONLINEAR PROGRAMMING*

ANDERS FORSGREN[†] AND PHILIP E. GILL[‡]

Abstract. This paper concerns large-scale general (nonconvex) nonlinear programming when first and second derivatives of the objective and constraint functions are available. A method is proposed that is based on finding an approximate solution of a sequence of unconstrained subproblems parameterized by a scalar parameter. The objective function of each unconstrained subproblem is an augmented penalty-barrier function that involves both primal and dual variables. Each subproblem is solved with a modified Newton method that generates search directions from a primal-dual system similar to that proposed for interior methods. The augmented penalty-barrier function may be interpreted as a merit function for values of the primal and dual variables.

An inertia-controlling symmetric indefinite factorization is used to provide descent directions and directions of negative curvature for the augmented penalty-barrier merit function. A method suitable for large problems can be obtained by providing a version of this factorization that will treat large sparse indefinite systems.

Key words. nonlinear programming, constrained minimization, primal-dual methods, interior methods, penalty methods, barrier methods, modified Newton methods

AMS subject classifications. 49M37, 65F05, 65K05, 90C30

PII. S1052623496305560

1. Introduction. This paper concerns second-derivative line-search methods for the solution of the nonlinear programming problem

$$\begin{array}{ll} \text{NP} & \begin{array}{l} \text{minimize} \quad f(x) \\ \text{subject to} \quad c_i(x) = 0, \quad i \in \mathcal{E}, \\ \quad \quad \quad c_i(x) \geq 0, \quad i \in \mathcal{I}, \end{array} \end{array}$$

where $c(x)$ is an m -vector of nonlinear functions with i th component $c_i(x)$, $i = 1, \dots, m$, and $\mathcal{E}$ and $\mathcal{I}$ are nonintersecting index sets. Throughout, it is assumed that f and c are twice-continuously differentiable, with the gradient of $f(x)$ denoted by $g(x)$ and the $m \times n$ Jacobian of $c(x)$ denoted by $J(x)$.

The methods considered in this paper are all *interior methods* in the sense that, given an initial iterate x_0 in the strict interior of the set $\{x \mid c_i(x) \geq 0, i \in \mathcal{I}\}$, then all subsequent iterates lie in the strict interior. In recent years, interior methods for NP have received considerable attention because of their close relationship with the “new” polynomial approaches to linear and quadratic programming. The methods of this paper exploit some of this recent research, particularly in the area of primal-dual methods for convex programming.

The format for the constraints allows a general inequality constraint to be treated either directly as an inequality $c_i(x) \geq 0$ or indirectly as an equality and nonnegative

*Received by the editors June 25, 1996; accepted for publication (in revised form) November 23, 1997; published electronically September 23, 1998.

<http://www.siam.org/journals/siopt/8-4/30556.html>

[†]Optimization and Systems Theory, Department of Mathematics, Royal Institute of Technology, SE-100 44 Stockholm, Sweden (andersf@math.kth.se). The research of this author was supported by the Swedish Natural Science Research Council (NFR) and the Fulbright Commission.

[‡]Department of Mathematics, University of California, San Diego, La Jolla, CA 92093-0112 (pgill@ucsd.edu). The research of this author was supported by National Science Foundation grant DMI-9424639 and Office of Naval Research grant N00014-96-1-0274.

slack variable $c_i(x) - s_i = 0$, $s_i \geq 0$. We make this distinction because the methods considered here treat equality and inequality constraints differently: inequality constraints are evaluated only at feasible points, whereas equality constraints are feasible only as a solution is approached. This implies that the equality constraint $c_i(x) - s_i = 0$ is likely to be evaluated at a point for which $c_i(x) < 0$. The only exception occurs with linear equality constraints, which can be kept feasible at every iterate (see, e.g., Gill, Murray, and Wright [19]).

In many physical and engineering applications, the constraint functions not only characterize the desired properties of the solution but also define a region in which the problem statement is meaningful (for example, $f(x)$ or some of the constraint functions may be undefined outside the feasible region). In these applications, an interior point can usually be determined trivially. Interior methods are therefore highly appropriate for this class of problem. Ironically, interior methods were originally proposed for precisely this reason, but this feature was largely overlooked during the revival of interior methods because of the emphasis on computational complexity. One aspect of our research on interior methods has concerned the benefits provided by the knowledge of strictly feasible approximate solutions. In certain applications, the exploitation of feasibility may account for the superiority of some specialized methods compared to generic NLP methods (e.g., the method of moving asymptotes in structural optimization; see Svanberg [40]). The availability of general-purpose methods that generate interior points will considerably strengthen our ability to solve practical problems efficiently.

The proposed algorithm for NP is based on solving a sequence of unconstrained minimization problems parameterized by a positive parameter μ . In the classical penalty-barrier method given by Fiacco and McCormick [13], the unconstrained function

$$(1.1) \quad M^\mu(x) = f(x) + \frac{1}{2\mu} \sum_{i \in \mathcal{E}} c_i(x)^2 - \mu \sum_{i \in \mathcal{I}} \ln c_i(x)$$

is minimized for a sequence of decreasing values of μ . The first constraint term on the right-hand side is the usual quadratic penalty function with penalty parameter $1/(2\mu)$. The second constraint term is the logarithmic barrier function, which creates a positive singularity at the boundary of the feasible region and thereby enforces strict feasibility while approaching the solution.

The penalty-barrier methods described here involve *outer* and *inner* iterations. Each outer iteration is associated with an element of a decreasing positive sequence of parameters $\{\mu_j\}$ such that $\lim_{j \rightarrow \infty} \mu_j = 0$. The inner iterations correspond to an iterative process for the unconstrained minimization of $M^\mu(x)$ for a given μ . The first-order optimality conditions for the unconstrained problem imply that there exist vectors $(x(\mu), \lambda(\mu))$ such that

$$(1.2) \quad g(x(\mu)) - J(x(\mu))^T \lambda(\mu) = 0, \quad \text{where } \lambda_i(\mu) = \begin{cases} -c_i(x(\mu))/\mu & \text{if } i \in \mathcal{E}, \\ \mu/c_i(x(\mu)) & \text{if } i \in \mathcal{I}. \end{cases}$$

The m -vector $\lambda(\mu)$ can be interpreted as an estimate of λ^* , the Lagrange multipliers of NP, and Fiacco and McCormick [13] give conditions under which local solutions $(x(\mu), \lambda(\mu))$ of $M^\mu(x)$ converge to (x^*, λ^*) as $\mu \rightarrow 0$. When regarded as a function of the parameter μ , the set of minimizers $x(\mu)$ defines a continuously differentiable path known as the *trajectory* or *penalty-barrier trajectory*.

The direct unconstrained minimization of $M^\mu(x)$ involves a number of serious difficulties that all result from the fact that the minimization problem becomes increasingly ill conditioned as μ is decreased. These difficulties have been addressed in recent papers by Gould [22, 23], Conn, Gould, and Toint [8], and Dussault [11], who propose several modifications of the classical penalty-barrier algorithm. As in the original method, these modifications emphasize the minimization of $M^\mu(x)$ with respect to the “primal” variables x and define new estimates of $\lambda(\mu)$ as a by-product of the main computation. Dussault [11] observes that these multiplier estimates can be inaccurate when x is not close to the trajectory and proposes computing least-squares multipliers from an estimate of the active constraints at x^* .

In this paper we propose a different approach in which estimates of $x(\mu)$ and $\lambda(\mu)$ are updated simultaneously. This approach has been very successful for interior methods for convex programming, but its extension to the nonconvex case has been problematic because of the lack of a suitable *merit function* or *potential function* that forces convergence from an arbitrary starting point. Here we propose a merit function based on the classical penalty-barrier function augmented by a weighted *proximity measure* that measures the distance of (x, λ) to the trajectory $(x(\mu), \lambda(\mu))$. This *augmented penalty-barrier function* is denoted by $M^{\mu, \nu}(x, \lambda)$, where ν denotes the positive weight on the proximity measure (see section 3.1). The function $M^{\mu, \nu}(x, \lambda)$ has several important properties.

- The penalty-barrier solutions $x(\mu)$ and $\lambda(\mu)$ are *local minimizers* of $M^{\mu, \nu}(x, \lambda)$.
- $M^{\mu, \nu}(x, \lambda)$ can be minimized with respect to both x and λ using a modified Newton method that generates search directions from a symmetric *primal-dual* system similar to that proposed for interior methods. These search directions can be calculated in a numerically stable way, even when the Hessian $\nabla^2 M^{\mu, \nu}(x, \lambda)$ is not positive definite.
- For all positive μ and ν , under certain assumptions, the inner iterates converge to a point $x(\mu)$ satisfying the second-order necessary conditions for a minimizer of $M^{\mu, \nu}(x, \lambda)$.

The paper is organized as follows. In section 2 we review some basic properties of the classical penalty-barrier method and consider some recent developments in primal-dual interior methods for convex programming. The unsymmetric and symmetric primal-dual equations and their properties are discussed in sections 2.3 and 2.4. In section 3 we propose a new augmented penalty-barrier merit function and provide a modified Newton method for minimizing this function with respect to both the primal and dual variables. The linear system associated with this method is shown to be equivalent to the symmetric form of the primal-dual equations. It follows that $M^{\mu, \nu}(x, \lambda)$ may be interpreted as a merit function for primal and dual variables generated by the primal-dual system. Finally, in section 4 we describe how the search direction can be calculated using a certain inertia-controlling symmetric indefinite factorization of the primal-dual system. This factorization allows the efficient calculation of descent directions and directions of negative curvature for the merit function. A method suitable for large problems can be obtained by providing a version of this factorization that will treat large sparse indefinite systems.

In practice, the penalty-barrier subproblem is never solved to completion. Instead, the subproblem is terminated early, and the next value of μ is chosen so that the sequence of aggregated inner iterates converges to x^* at a rapid rate. However, although our choice of merit function is influenced by the desire for rapid local convergence, the principal focus is on the construction of the inner iterations, i.e., on the

minimization of $M^{\mu,\nu}(x, \lambda)$ for given μ and ν . Methods that terminate the subproblems early and define an associated decrease in μ will be considered in a future paper. Such methods rely heavily on the techniques discussed here.

2. Background.

2.1. Nomenclature. Given a real symmetric matrix A , the inertia of A —denoted by $In(A)$ —is the associated integer triple (a_+, a_-, a_0) indicating the number of positive, negative, and zero eigenvalues of A . The largest and smallest eigenvalues of A are denoted by $\eta_{\max}(A)$ and $\eta_{\min}(A)$. The expression $A \succ 0$ means that A is positive definite. A member of a sequence of real symmetric matrices $\{A_k\}$ is said to be sufficiently positive definite if its eigenvalues are greater than a small positive number independent of k . The vector e_i is used to denote the i th unit vector of the appropriate dimension. The vector e denotes the vector of all ones whose dimension is defined by the context of its use.

2.2. The classical penalty-barrier function. Standard second-order line-search or trust-region methods are easily adapted to find a local minimizer of M^μ (for details see, e.g., Dennis and Schnabel [10]). Regardless of the choice of method, as the iterates converge to an isolated local solution $x(\mu)$, the change in variables at iterate x is determined by the Newton equations $\nabla^2 M^\mu(x) \Delta x = -\nabla M^\mu(x)$, which define Δx as the step to the point that minimizes a quadratic model of M^μ . The Hessian $\nabla^2 M^\mu(x)$ and gradient $\nabla M^\mu(x)$ can be represented in terms of auxiliary quantities $\pi^\mu(x)$ and $\Omega^\mu(x)$ as follows. Let $\pi^\mu(x)$ denote the m -vector:

$$(2.1) \quad \pi_i^\mu(x) = \begin{cases} -c_i(x)/\mu & \text{if } i \in \mathcal{E}, \\ \mu/c_i(x) & \text{if } i \in \mathcal{I}. \end{cases}$$

For any given μ and x , the vector $\pi^\mu(x)$ approximates $\lambda(\mu)$ (and hence λ^*). In particular, if $x = x(\mu)$, then $\pi^\mu(x) = \lambda(\mu)$, from (1.2).

Let $\Omega^\mu(x)$ denote the diagonal matrix $\text{diag}(\omega^\mu(x))$, where

$$(2.2) \quad \omega_i^\mu(x) = \begin{cases} 1/\mu & \text{if } i \in \mathcal{E}, \\ \mu/c_i(x)^2 & \text{if } i \in \mathcal{I}. \end{cases}$$

Given $\pi^\mu(x)$ and $\Omega^\mu(x)$, the derivatives of M^μ can be written as

$$(2.3a) \quad \nabla M^\mu(x) = g(x) - J(x)^T \pi^\mu(x),$$

$$(2.3b) \quad \nabla^2 M^\mu(x) = H(x, \pi^\mu(x)) + J(x)^T \Omega^\mu(x) J(x),$$

where $H(x, \lambda)$ denotes the Hessian with respect to x of the Lagrangian $f(x) - \lambda^T c(x)$.

Unconstrained methods for minimizing M^μ are well defined as long as the derivatives (2.3) are bounded. This will always be the case if c is evaluated at points that are strictly feasible with respect to the inequality constraints. For example, in a line-search method, the initial estimate x_0 is chosen to lie in the strict interior of the region $\{x \mid c_i(x) \geq 0, i \in \mathcal{I}\}$, and subsequent line searches ensure that $c_i(x + \alpha \Delta x) > 0$ for all $i \in \mathcal{I}$ (see, e.g., Murray and Wright [37]).

Given the derivatives (2.3), the Newton equations can be written as

$$(2.4) \quad \left(H(x, \pi^\mu(x)) + J(x)^T \Omega^\mu(x) J(x) \right) \Delta x = -(g(x) - J(x)^T \pi^\mu(x)).$$

Unfortunately, the direct minimization of M^μ cannot be recommended as a practical algorithm. As $\mu \rightarrow 0$, terms in the matrix $J(x)^T \Omega^\mu(x) J(x)$ associated with the gradients of the equality constraints and active inequality constraints can become unbounded. If the number $\bar{m}$ of unbounded terms satisfies $0 < \bar{m} < n$, then $\nabla^2 M^\mu(x)$ becomes increasingly ill conditioned as μ is reduced, and the inverse of $\nabla^2 M^\mu(x)$ is singular in the limit (see Murray [36] and Wright [41]).

The ill-conditioning of $\nabla^2 M^\mu$ not only affects the sensitivity of the numerical solution of (2.4) but also inhibits the final rate of convergence of the Newton iterates. In the next three sections we review some recent approaches that are intended to overcome these difficulties.

2.3. The primal-dual equations. The effects of inevitable ill-conditioning can be removed by exploiting some recent developments in interior methods for convex programming. These methods compute primal and dual variables that satisfy a system of nonlinear equations equivalent to a perturbed system of first-order optimality conditions.

For any value of μ , an associated point $(x(\mu), \lambda(\mu))$ on the penalty-barrier trajectory satisfies the $n + m$ equations

$$(2.5a) \quad g(x) - J(x)^T \lambda = 0,$$

$$(2.5b) \quad \mu \lambda_i + c_i(x) = 0, \quad i \in \mathcal{E},$$

$$(2.5c) \quad c_i(x) \lambda_i - \mu = 0, \quad i \in \mathcal{I}.$$

These relations imply that $(x(\mu), \lambda(\mu))$ can be determined by solving $n + m$ nonlinear equations in the $n + m$ unknowns (x, λ) using an iterative method, which is usually some variant of Newton's method.

To derive a suitable form of Newton's method, it is helpful to rewrite the m relations (2.5b)–(2.5c) in vector form $W^\mu(x)(\lambda - \pi^\mu(x)) = 0$, where $\pi^\mu(x)$ is given by (2.1) and $W^\mu(x)$ is the diagonal matrix with diagonal entries

$$(2.6) \quad w_i^\mu(x) = \begin{cases} \mu & \text{if } i \in \mathcal{E}, \\ c_i(x) & \text{if } i \in \mathcal{I}. \end{cases}$$

Conditions (2.5) imply that $(x(\mu), \lambda(\mu))$ solves the $n + m$ nonlinear equations $F^\mu(x, \lambda) = 0$, where $F^\mu(x, \lambda)$ and its Jacobian $F^{\mu'}(x, \lambda)$ are given by

$$F^\mu(x, \lambda) = \begin{pmatrix} g(x) - J(x)^T \lambda \\ W^\mu(x)(\lambda - \pi^\mu(x)) \end{pmatrix}, \quad F^{\mu'}(x, \lambda) = \begin{pmatrix} H(x, \lambda) & -J(x)^T \\ Z(\lambda)J(x) & W^\mu(x) \end{pmatrix},$$

and $Z(\lambda)$ is a diagonal matrix with diagonal entries

$$(2.7) \quad z_i(\lambda) = \begin{cases} 1 & \text{if } i \in \mathcal{E}, \\ \lambda_i & \text{if } i \in \mathcal{I}. \end{cases}$$

Suppose that (x, λ) is an estimate of $(x(\mu), \lambda(\mu))$. Let π , g , c , J , H , and W denote the quantities $\pi^\mu(x)$, $g(x)$, $c(x)$, $J(x)$, $H(x, \lambda)$, and $W^\mu(x)$. Given (x, λ) , the next iterate of Newton's method for a solution of $F^\mu = 0$ is $(x + \Delta x, \lambda + \Delta \lambda)$, where $(\Delta x, \Delta \lambda)$ satisfies the Newton equations

$$(2.8) \quad \begin{pmatrix} H & -J^T \\ ZJ & W \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta \lambda \end{pmatrix} = - \begin{pmatrix} g - J^T \lambda \\ W(\lambda - \pi) \end{pmatrix}.$$

Similar equations have been proposed for interior methods for convex programming (see, e.g., Kojima, Mizuno, and Yoshise [28], Lustig, Marsten, and Shanno [29], Megiddo [31], Mehrotra [32], Monteiro and Adler [33], and Gill et al. [18]) and non-convex programming (see, e.g., El-Bakry et al. [12] and Forsgren, Gill, and Shinnerl [16]).

The class of methods considered in this paper is characterized by the property that every iterate (x, λ) satisfies $c_i(x) > 0$ and $\lambda_i > 0$ for all $i \in \mathcal{I}$. (It is this property that has come to be associated with the idea of a “primal-dual” method.) If this property is satisfied at an initial point (x_0, λ_0) , then it can be enforced at every subsequent iterate if we are prepared to restrict the step along each search direction. In this case, the next iterate will be $(x + \alpha \Delta x, \lambda + \alpha \Delta \lambda)$, where α is a step length such that $c_i(x + \alpha \Delta x) > 0$ and $\lambda_i + \alpha \Delta \lambda_i > 0$ for all $i \in \mathcal{I}$. (The primal-dual property is enforced automatically by the merit function proposed in section 3.1.) With this restriction on the choice of x and λ , the Newton equations (2.8) are known as the *unsymmetric primal-dual equations*. Note that the diagonal matrices Z and W are positive definite.

The primal-dual equations have the important property that, if the second-order sufficient conditions for optimality hold, the Jacobian $F^{\mu'}(x, \lambda)$ is nonsingular in a neighborhood of (x^*, λ^*) for μ sufficiently small (see Fiacco and McCormick [13]). This result has two important implications. First, it allows the application of the standard convergence theory for Newton’s method, which states that the Newton iterates will converge quadratically in a sufficiently small neighborhood of $(x(\mu), \lambda(\mu))$ for μ sufficiently small.

The second benefit of a nonsingular primal-dual Jacobian concerns the quality of the first Newton iterate immediately after μ is reduced (to $\bar{\mu}$, say). In practice, each unconstrained minimization is done only approximately, with a termination condition that determines the accuracy of the unconstrained minimizers. An important feature of any implementation of a penalty-barrier method is that, in the limit, this termination condition should allow only one inner iteration for each value of μ . If this is to be the case, the solution of the first primal-dual system associated with $\bar{\mu}$ must provide an increasingly good estimate of $(x(\bar{\mu}), \lambda(\bar{\mu}))$. If second-order sufficiency holds at (x^*, λ^*) , and μ is reduced to $\bar{\mu}$ at a point $(x(\mu), \lambda(\mu))$ on the trajectory, then the new primal-dual nonlinear system will be “close” to the old one for μ sufficiently small. This implies that the aggregated set of inner iterates will converge very rapidly, given a suitable scheme for reducing μ .

Alternative methods for $(x(\mu), \lambda(\mu))$ can be derived by linearizing equivalent forms of the perturbed condition (2.5c) (see, e.g., Ye [45], Gill et al. [18], and Gonzaga [21]). However, it must be emphasized that the favorable property mentioned above does not hold for these alternatives. For example, suppose that (2.5c) is replaced by the condition $\lambda_i - \mu/c_i(x) = 0$ for $i \in \mathcal{I}$; or, equivalently, $\lambda_i - \pi^\mu(x) = 0$ for $i \in \mathcal{I}$. The associated Newton equations for the combined constraints are

$$(2.9) \quad \begin{pmatrix} H & -J^T \\ \Omega J & I \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta \lambda \end{pmatrix} = - \begin{pmatrix} g - J^T \lambda \\ \lambda - \pi \end{pmatrix},$$

where Ω is the diagonal matrix $\Omega^\mu(x)$ of (2.2). These equations are known as the *unsymmetric primal barrier equations* for (x, λ) . If these equations are defined with $\lambda = \pi^\mu(x)$, and $\Delta \lambda$ is eliminated from the system, we obtain the classical penalty-barrier equations (2.4) for Δx . This implies that the solution of (2.9) can be used

to find a solution of (2.4) in a numerically stable way (see Gould [22], who first proposed this calculation for the corresponding row-scaled symmetric system). However, although the Newton direction can be calculated accurately, the Jacobian associated with the primal system (2.9) is unbounded as $\mu \rightarrow 0$, and the region of quadratic convergence shrinks with μ . Moreover, reducing μ may give a new Δx that differs substantially from the step to the minimizer on the trajectory (see Wright [42] for a discussion of this point in the context of the primal barrier method). For a discussion of the asymptotic rate of convergence of the classical penalty-barrier method, see Gould [23] and Dussault [11]. They show that for certain termination criteria, an initial inner iterate can be calculated so that the aggregated sequence of inner iterates converges at a two-step Q -quadratic rate. Strategies for modifying the classical barrier method are considered by Conn, Gould, and Toint [8].

2.4. The symmetric primal-dual system. An important implication of solving nonconvex problems is the prominent role of *symmetric* systems in the definition of the search direction. If the objective is not convex, even the verification of optimality requires knowledge of the inertia (i.e., the number of positive, negative, and zero eigenvalues) of the Hessian in the subspace orthogonal to the active constraint gradients. It is not at all obvious how the inertia of a matrix can be estimated efficiently without utilizing its symmetry.

The unsymmetric primal-dual equations can be symmetrized by premultiplying the last m rows by Z^{-1} and changing the sign of $\Delta\lambda$. This gives

$$(2.10) \quad \begin{pmatrix} H & J^T \\ J & -D \end{pmatrix} \begin{pmatrix} \Delta x \\ -\Delta\lambda \end{pmatrix} = - \begin{pmatrix} g - J^T\lambda \\ D(\lambda - \pi) \end{pmatrix},$$

where $D = Z^{-1}W$. Following the convention used in convex programming, we call these equations the *symmetric primal-dual equations*.

The diagonal matrix D in (2.10) plays a crucial role in the analysis. Formally, we write $D = D^\mu(x, \lambda)$, with

$$(2.11) \quad d_i^\mu(x, \lambda) = \begin{cases} \mu & \text{if } i \in \mathcal{E}, \\ c_i(x)/\lambda_i & \text{if } i \in \mathcal{I}. \end{cases}$$

If the second-order sufficient conditions for optimality hold at (x^*, λ^*) , then the elements of D^μ either go to zero or go to infinity as $\mu \rightarrow 0$. For equality constraints and active inequality constraints, $d_i^\mu(x(\mu), \lambda(\mu)) \rightarrow 0$, and for inactive inequality constraints, $d_i^\mu(x(\mu), \lambda(\mu)) \rightarrow \infty$. This implies that the symmetric system (2.10) is ill conditioned, with some diagonal elements becoming unbounded as $\mu \rightarrow 0$. Forsgren, Gill, and Shinnerl [16] show that this ill-conditioning is artificial in the sense that the true sensitivity of the solution is independent of the size of the large diagonals. In particular, they show that the primal-dual solution can be found accurately using a certain symmetric indefinite factorization of the primal-dual system (see section 4.1). Related stability discussions can also be found in Poncelaón [38] and Wright [43, 44].

2.5. Global convergence. Until now we have considered the *local* properties of the primal-dual method. In doing so, we have made two assumptions: (i) the iterates (x, λ) lie in a sufficiently small neighborhood of $(x(\mu), \lambda(\mu))$ for μ sufficiently small; and (ii) $(x(\mu), \lambda(\mu))$ lies on a trajectory of *local minimizers* of M^μ . In order to make the method suitable for general problems, it is necessary to use the primal-dual direction $(\Delta x, \Delta\lambda)$ in conjunction with a merit function that forces convergence from

an arbitrary starting point. Moreover, since second derivatives of f and c are being calculated at each iteration, it seems reasonable to expect that every limit point of the sequence of iterates should satisfy the *second-order* necessary conditions for optimality.

In the case of primal-dual *interior* methods (where it is assumed that all constraints are inequalities), various merit functions have been suggested. One particularly popular merit function is

$$M(x, \lambda) = \frac{1}{2} F^\mu(x, \lambda)^T F^\mu(x, \lambda),$$

where $F^\mu(x, \lambda)$ is the residual vector associated with the primal-dual equations (2.10). El-Bakry et al. [12] propose this merit function for general nonlinear programming and show that, under certain assumptions, it provides convergence to a point that satisfies the first-order conditions for optimality. In section 3.1, we propose a merit function with the potential of giving limit points that satisfy the second-order necessary optimality conditions.

3. Second-order penalty-barrier methods.

3.1. An augmented penalty-barrier merit function. For positive parameters μ and ν , consider the combined *augmented penalty and barrier function* $M^{\mu, \nu}(x, \lambda)$ such that

$$(3.1) \quad \begin{aligned} M^{\mu, \nu}(x, \lambda) = & f(x) + \frac{1}{2\mu} \sum_{i \in \mathcal{E}} (c_i(x)^2 + \nu(c_i(x) + \mu\lambda_i)^2) \\ & - \mu \sum_{i \in \mathcal{I}} \left(\ln c_i(x) + \nu \left(\ln \left(\frac{c_i(x)\lambda_i}{\mu} \right) + 1 - \frac{c_i(x)\lambda_i}{\mu} \right) \right). \end{aligned}$$

This function is well defined for all (x, λ) such that $c_i(x) > 0$ and $\lambda_i > 0$ for $i \in \mathcal{I}$.

For fixed x , the vector $\lambda = \pi^\mu(x)$ minimizes $M^{\mu, \nu}(x, \lambda)$ with respect to λ , where $\pi^\mu(x)$ is defined in (2.1). Moreover, $M^{\mu, \nu}(x, \pi^\mu(x)) = M^\mu(x)$, where $M^\mu(x)$ denotes the classical penalty-barrier function (1.1). This result implies that a point $(x(\mu), \lambda(\mu))$ on the penalty-barrier trajectory can be found by minimizing $M^{\mu, \nu}(x, \lambda)$ with respect to both x and λ . The following lemma makes this precise.

LEMMA 3.1. *Let $M^{\mu, \nu}(x, \lambda)$ be the augmented penalty-barrier function (3.1) defined with any positive μ and ν . A point (x, λ) such that $c_i(x) > 0$ and $\lambda_i > 0$ for $i \in \mathcal{I}$ is an unconstrained local minimizer of $M^{\mu, \nu}(x, \lambda)$ if and only if $(x, \lambda) = (x(\mu), \lambda(\mu))$. Furthermore, $\min_\lambda M^{\mu, \nu}(x, \lambda) = M^{\mu, \nu}(x, \pi^\mu(x)) = M^\mu(x)$, where $M^\mu(x)$ is the classical combined penalty-barrier function (1.1) and $\pi^\mu(x)$ is defined in (2.1).*

Proof. For x fixed, $M^{\mu, \nu}(x, \lambda)$ is a strictly convex separable function of λ for $\lambda > 0$. It is straightforward to verify that the minimizing λ is given by $\pi^\mu(x)$, where $\pi^\mu(x)$ is given by (2.1). Substituting these values in (3.1) gives $M^{\mu, \nu}(x, \pi^\mu(x)) = M^\mu(x)$, where $M^\mu(x)$ is given by (1.1). The definition of $x(\mu)$ as a minimizer of $M^\mu(x)$ in conjunction with the definition $\lambda(\mu) = \pi^\mu(x(\mu))$ gives the required result. $\square$

The augmented penalty-barrier function $M^{\mu, \nu}(x, \lambda)$ can be interpreted as the classical combined penalty-barrier function augmented by a function that measures the proximity of (x, λ) to the penalty-barrier trajectory. The parameter ν defines the relative weight of the proximity measure.

The most important property of $M^{\mu, \nu}(x, \lambda)$ is that it is *minimized* at a point $(x(\mu), \lambda(\mu))$ on the trajectory. (We observe that this is in contrast to the saddle-point property typically associated with Lagrangians and augmented Lagrangians.)

It follows that the Fiacco–McCormick penalty-barrier method can be generalized to include dual variables by minimizing $M^{\mu,\nu}(x, \lambda)$ with respect to both x and λ using any unconstrained minimization method. In this case, a second-order method will require both the first and second derivatives of $M^{\mu,\nu}(x, \lambda)$. Differentiating $M^{\mu,\nu}(x, \lambda)$ with respect to both x and λ and using the auxiliary quantities $\pi^\mu(x)$ (2.1) and $D^\mu(x, \lambda)$ (2.11) give

$$(3.2a) \quad \nabla M^{\mu,\nu}(x, \lambda) = \begin{pmatrix} g(x) - (1 + \nu)J(x)^T \pi^\mu(x) + \nu J(x)^T \lambda \\ \nu D^\mu(x, \lambda)(\lambda - \pi^\mu(x)) \end{pmatrix}, \quad \text{and}$$

$$(3.2b) \quad \nabla^2 M^{\mu,\nu}(x, \lambda) = \begin{pmatrix} \nabla_{xx}^2 M^{\mu,\nu}(x, \lambda) & \nu J(x)^T \\ \nu J(x) & \nu \Gamma^\mu(\lambda) \end{pmatrix},$$

where $\Gamma^\mu(\lambda)$ is the diagonal matrix $\text{diag}(\gamma^\mu(\lambda))$ such that

$$(3.3) \quad \gamma_i^\mu(\lambda) = \begin{cases} \mu & \text{if } i \in \mathcal{E}, \\ \mu/\lambda_i^2 & \text{if } i \in \mathcal{I}. \end{cases}$$

The Hessian $\nabla_{xx}^2 M^{\mu,\nu}(x, \lambda)$ is written in terms of the Lagrangian Hessian $H(x, \lambda)$. Differentiating $M^{\mu,\nu}(x, \lambda)$ twice with respect to x and collecting terms gives

$$\nabla_{xx}^2 M^{\mu,\nu}(x, \lambda) = H\left(x, (1 + \nu)\pi^\mu(x) - \nu\lambda\right) + (1 + \nu)J(x)^T \Omega^\mu(x) J(x),$$

where $\Omega^\mu(x)$ is defined in (2.2).

Rather than use these derivatives directly within a “black-box” line-search or trust-region method, we propose that $\nabla^2 M^{\mu,\nu}(x, \lambda)$ be *approximated* by a symmetric matrix $S^{\mu,\nu}(x, \lambda)$ in such a way that the search direction is defined by the primal-dual equations (2.10). This strategy requires the use of a modified Newton method that can exploit an approximate Hessian. Such methods are described next.

3.2. Modified Newton methods using an approximate Hessian. We consider a certain class of modified Newton methods for the minimization of a function $M(v)$ with gradient $\nabla M(v)$ and Hessian $\nabla^2 M(v)$. For brevity, we consider a method similar to that discussed by Forsgren, Gill, and Murray [15]. However, the modified Newton methods of McCormick [30], Mukai and Polak [35], Kaniel and Dax [27], Moré and Sorensen [34], and Goldfarb [20] are all suitable alternatives.

The method generates a sequence $\{v_k\}_{k=0}^\infty$ of improving estimates of a local minimizer of M . At iteration k , a line search is performed along a nonzero direction Δv_k of the form

$$\Delta v_k = s_k + d_k,$$

where s_k and d_k are computed using a symmetric matrix $S(v)$ that approximates the Hessian $\nabla^2 M(v)$ at v_k . The crucial property of S is that the approximation is exact at any stationary point of M . In particular, we require that

$$\|S(v_k) - \nabla^2 M(v_k)\| = O(\|v_k - v^*\|),$$

where v^* is any stationary point of M . We emphasize that the exact Hessian $\nabla^2 M(v)$ is known at each step, but that we choose not to use it to compute the search direction (see section 3.1). Note that a regular modified Newton method is obtained if $S(v) = \nabla^2 M(v)$ for all v .

Although s_k and d_k are computed using $S(v_k)$, they are forced to have certain properties with respect to $\nabla M(v_k)$ and $S(v_k)$. In particular, we ensure that $\nabla M(v_k)^T s_k \leq 0$, $\nabla M(v_k)^T d_k \leq 0$, and $d_k^T S(v_k) d_k \leq 0$. Each s_k satisfies a positive-definite system of the form

$$\bar{S}(v_k) s_k = -\nabla M(v_k),$$

where $\bar{S}(v_k) = S(v_k) + E_k$ with $E_k = 0$ if $S(v_k)$ is sufficiently positive definite. The vector d_k is defined as a direction of negative curvature for $S(v_k)$ and is normalized so that $\nabla M(v_k)^T d_k \leq 0$. If $d_k^T \nabla^2 M(v_k) d_k > 0$, the vector d_k can be reset to zero. This is not essential, but it provides the convenient property that $\nabla^2 M(v_k)$ has at least one zero or negative eigenvalue if d_k is nonzero.

Once Δv_k has been determined, a line search is used to find a step length α_k such that $v_{k+1} = v_k + \alpha_k \Delta v_k$ and $M(v_k + \alpha_k \Delta v_k) < M(v_k)$. The principal role of the line search is to ensure that $\nabla M(v_k)^T s_k \rightarrow 0$ and $d_k^T \nabla^2 M(v_k) d_k \rightarrow 0$. If these conditions are satisfied, and the directions s_k and d_k are *sufficient* in the sense that $\{s_k\}$ and $\{d_k\}$ are bounded sequences that satisfy the conditions

$$(3.4a) \quad \nabla M(v_k)^T s_k \rightarrow 0 \Rightarrow \nabla M(v_k) \rightarrow 0 \quad \text{and} \quad s_k \rightarrow 0,$$

$$(3.4b) \quad d_k^T \nabla^2 M(v_k) d_k \rightarrow 0 \Rightarrow \min\{\eta_{\min}(\nabla^2 M(v_k)), 0\} \rightarrow 0 \quad \text{and} \quad d_k \rightarrow 0,$$

then every limit point of the sequence $\{v_k\}_{k=0}^\infty$ will satisfy the second-order necessary conditions for optimality (see Moré and Sorensen [34]). In our case, conditions are imposed on $d_k^T S(v_k) d_k$ rather than $d_k^T \nabla^2 M(v_k) d_k$. However, the properties of $S(v_k)$ ensure that $d_k^T S(v_k) d_k \rightarrow d_k^T \nabla^2 M(v_k) d_k$ if $\nabla M(v_k) \rightarrow 0$, and it follows that, in the limit, $S(v_k)$ is positive semidefinite if and only if $\nabla^2 M(v_k)$ is positive semidefinite.

To keep things simple, a backtracking line search is described here. (For a review of some alternatives, see Moré and Sorensen [34].) An initial step $\alpha_k = 1$ is reduced (if necessary) by a constant factor until the reduction in M is at least as large as a fixed factor τ ($0 < \tau < 1$) of the reduction predicted by a model function formed from the first two or three terms of a Taylor-series approximation to $M(v_k + \alpha \Delta v_k)$. If $m_k(\alpha)$ denotes the univariate function $M(v_k + \alpha \Delta v_k)$, then a step length α_k is accepted if

$$m_k(\alpha) \leq m_k(0) + \tau \left(\alpha_k m'_k(0) + \frac{1}{2} \alpha_k^2 \min\{0, m''_k(0)\} \right),$$

where m' and m'' denote the first and second derivatives of $m(\alpha)$ with respect to α .

The modified Newton method can be summarized as follows.

ALGORITHM 3.1. *Modified Newton method.*

Specify τ such that $0 < \tau < \frac{1}{2}$;

Choose v_0 ;

$k \leftarrow 0$;

repeat

 Evaluate $M(v_k)$, $\nabla M(v_k)$;

if $S(v_k)$ is positive semidefinite and $\nabla M(v_k) = 0$ **then**

 stop; [v_k satisfies the second-order necessary conditions for optimality.]

else if $S(v_k)$ is positive semidefinite **then**

 Compute s_k such that $\nabla M(v_k)^T s_k < 0$; $d_k \leftarrow 0$;

else

 [$S(v_k)$ must have at least one negative eigenvalue.]

 Compute s_k such that $\nabla M(v_k)^T s_k \leq 0$;

Compute d_k such that $\nabla M(v_k)^T d_k \leq 0$ and $d_k^T S(v_k) d_k < 0$;
end
 $\Delta v_k \leftarrow s_k + d_k$;
 $\alpha_k \leftarrow 1$; $m_k(\alpha_k) \leftarrow M(v_k + \alpha_k \Delta v_k)$;
do while $m_k(\alpha_k) > m_k(0) + \tau(\alpha_k m'_k(0) + \frac{1}{2} \alpha_k^2 \min\{0, m''_k(0)\})$
 $\alpha_k \leftarrow \alpha_k/2$; $m_k(\alpha_k) \leftarrow M(v_k + \alpha_k \Delta v_k)$;
end do
 $v_{k+1} \leftarrow v_k + \alpha_k \Delta v_k$;
 $k \leftarrow k + 1$;
until converged;

3.3. The primal-dual method. Our aim is to use the primal-dual equations (2.10) to define the search direction but to relate this search direction to the augmented penalty-barrier merit function (3.1).

Consider the matrix $S^{\mu,\nu}(x, \lambda)$ obtained by approximating $\pi^\mu(x)$ by λ , $\Omega^\mu(x)$ by $D^\mu(x, \lambda)^{-1}$, and $\Gamma^\mu(\lambda)$ by $D^\mu(x, \lambda)$ in the matrix $\nabla^2 M^{\mu,\nu}(x, \lambda)$ of (3.2b). Then

$$(3.5) \quad S^{\mu,\nu}(x, \lambda) = \begin{pmatrix} H(x, \lambda) + (1 + \nu)J(x)^T D^\mu(x, \lambda)^{-1} J(x) & \nu J(x)^T \\ \nu J(x) & \nu D^\mu(x, \lambda) \end{pmatrix},$$

and $S^{\mu,\nu}(x, \lambda) \rightarrow \nabla^2 M^{\mu,\nu}(x, \lambda)$ as $(x, \lambda) \rightarrow (x(\mu), \lambda(\mu))$. It follows that $S^{\mu,\nu}(x, \lambda)$ can be used as an approximate Hessian in a modified Newton method of the type suggested in section 3.2.

The next result provides the theoretical basis for the new method. It indicates that the modified Newton direction based on the approximate Hessian $S^{\mu,\nu}(x, \lambda)$,

$$(3.6) \quad S^{\mu,\nu}(x, \lambda) \begin{pmatrix} \Delta x \\ \Delta \lambda \end{pmatrix} = -\nabla M^{\mu,\nu}(x, \lambda),$$

satisfies the primal-dual equations (2.8) or, equivalently, the symmetric primal-dual equations (2.10). Moreover, the approximation $S^{\mu,\nu}(x, \lambda)$ is exact whenever (x, λ) is a stationary point of $M^{\mu,\nu}$, i.e., when $\nabla M^{\mu,\nu}(x, \lambda) = 0$.

THEOREM 3.2. *Let $M^{\mu,\nu}(x, \lambda)$ be the augmented penalty-barrier function (3.1) defined with any positive μ and ν . Let $S^{\mu,\nu}(x, \lambda)$ denote the approximate Hessian (3.5). If $c_i(x) > 0$ and $\lambda_i > 0$ for all $i \in \mathcal{I}$, then the vector $(\Delta x, \Delta \lambda)$ solves the modified Newton equations (3.6) if and only if it solves the primal-dual equations (2.8) and the symmetric primal-dual equations (2.10). Moreover, if $\nabla M^{\mu,\nu}(x, \lambda) = 0$, then $S^{\mu,\nu}(x, \lambda) = \nabla^2 M^{\mu,\nu}(x, \lambda)$.*

Proof. If $c_i(x) > 0$ and $\lambda_i > 0$ for $i \in \mathcal{I}$, and $\mu > 0$, the diagonal matrices $D^\mu(x, \lambda)$ (2.11) and $Z(\lambda)$ (2.7) are positive definite. The definiteness of $Z(\lambda)$ implies that the unsymmetric primal-dual equations (2.8) and the symmetric primal-dual equations (2.10) are equivalent. Hence, it suffices to consider the symmetric equations (2.10). To simplify the notation, let $H = H(x, \lambda)$, $D = D^\mu(x, \lambda)$, $J = J(x)$, $g = g(x)$, and $\pi = \pi^\mu(x, \lambda)$. As ν is nonzero and D is positive definite, the $(n + m) \times (n + m)$ block upper-triangular matrix

$$\begin{pmatrix} I & (1 + \nu)J^T D^{-1} \\ 0 & \nu I \end{pmatrix}$$

is well defined, nonsingular, and can be applied to both sides of the modified Newton equation. The first part of the theorem follows directly from the identities

$$\begin{pmatrix} I & (1+\nu)J^TD^{-1} \\ 0 & \nu I \end{pmatrix} \begin{pmatrix} H & -J^T \\ J & D \end{pmatrix} = \begin{pmatrix} H + (1+\nu)J^TD^{-1}J & \nu J^T \\ \nu J & \nu D \end{pmatrix}$$

and

$$\begin{pmatrix} I & (1+\nu)J^TD^{-1} \\ 0 & \nu I \end{pmatrix} \begin{pmatrix} g - J^T\lambda \\ D(\lambda - \pi) \end{pmatrix} = \begin{pmatrix} g - (1+\nu)J^T\pi + \nu J^T\lambda \\ \nu D(\lambda - \pi) \end{pmatrix},$$

upon comparison of (2.10), (3.2a), (3.5), and (3.6).

For the second part of the proof, assume that $\nabla M^{\mu,\nu}(x, \lambda)$ is zero, with $c_i(x) > 0$ and $\lambda_i > 0$ for all $i \in \mathcal{I}$. It follows that $D^\mu(x, \lambda)$ (2.11) is nonsingular. Hence, (3.2a) implies that $\lambda = \pi^\mu(x)$, where $\pi^\mu(x)$ is defined by (2.1). It is then straightforward to verify that $\Gamma^\mu(\pi^\mu(x)) = D^\mu(x, \pi^\mu(x))$ and $\Omega^\mu(x) = D^\mu(x, \pi^\mu(x))^{-1}$. $\square$

It is essential that $S^{\mu,\nu}$ is an exact approximation of $\nabla^2 M^{\mu,\nu}$ at every stationary point of $M^{\mu,\nu}(x, \lambda)$. Broadly speaking, it is not necessary to compute a direction of negative curvature for $M^{\mu,\nu}$ at every point at which $\nabla^2 M^{\mu,\nu}$ is indefinite, but a direction of negative curvature must always be computed in a neighborhood of a stationary point at which the Hessian is indefinite.

4. Calculation of the search direction. Based on the discussion of the previous section, we define the search direction using the primal-dual matrix K such that

$$(4.1) \quad K = \begin{pmatrix} H & J^T \\ J & -D \end{pmatrix},$$

but use the matrix S such that

$$(4.2) \quad S = \begin{pmatrix} H + (1+\nu)J^TD^{-1}J & \nu J^T \\ \nu J & \nu D \end{pmatrix}$$

to measure the approximate curvature of the merit function. In the modified Newton method of section 3.2 the search direction is given by

$$\begin{pmatrix} \Delta x \\ \Delta \lambda \end{pmatrix} = \begin{pmatrix} s_x \\ s_\lambda \end{pmatrix} + \begin{pmatrix} d_x \\ d_\lambda \end{pmatrix} = s + d,$$

where s is a descent direction and d is a direction of negative curvature for the merit function (see section 3.2). Whenever the Hessian approximation S of (4.2) is sufficiently positive definite, s is computed from the primal-dual equations (2.10), and d is zero since S is positive definite.

Note that the inertia of S is needed at each step, yet it is the matrix K (4.1) that is being factorized. This makes it essential that the inertia of S can be deduced from the inertia of K . The following lemma gives the required relationships.

LEMMA 4.1. *Let D be a symmetric positive-definite $m \times m$ matrix, let H be a symmetric $n \times n$ matrix, let ν be positive, let K be the primal-dual KKT matrix (4.1), and let S be the approximate Hessian (4.2). Then*

$$\begin{aligned} \text{In}(S) &= \text{In}(H + J^TD^{-1}J) + (m, 0, 0), \\ \text{In}(K) &= \text{In}(H + J^TD^{-1}J) + (0, m, 0). \end{aligned}$$

Moreover, if $H + J^T D^{-1} J$ has at least one negative eigenvalue, then

$$\eta_{\min}(H + J^T D^{-1} J) \leq \eta_{\min}(S).$$

Proof. Let X be a symmetric block-partitioned matrix of the form

$$X = \begin{pmatrix} A & B^T \\ B & C \end{pmatrix}.$$

The basic properties of the inertia give $\text{In}(X) = \text{In}(C) + \text{In}(A - B^T C^{-1} B)$; see, e.g., Cottle [9, p. 197]. Furthermore, if $A - B^T C^{-1} B$ has at least one negative eigenvalue and C is positive definite, then

$$\eta_{\min}(A - B^T C^{-1} B) \leq \eta_{\min}(X)$$

(see Lemma 2.1 in Forsgren, Gill, and Murray [15]).

The inertia property implies that $\text{In}(S) = \text{In}(\nu D) + \text{In}(H + J^T D^{-1} J)$. Since νD is a positive diagonal, $\text{In}(\nu D) = (m, 0, 0)$ and the first result follows. Similarly, $\text{In}(K) = \text{In}(-D) + \text{In}(H + J^T D^{-1} J)$. Since $\text{In}(-D) = (0, m, 0)$, the second result also holds. Finally, the eigenvalue relation follows from the above relation with $X = S$ and $C = \nu D$. $\square$

4.1. The factorization. We assume that the primal-dual KKT equations are solved using a variant of the *symmetric indefinite factorization* (see Bunch and Parlett [6], Fletcher [14], and Bunch and Kaufman [4]), which we refer to as the *LBL^T factorization*. If K denotes the particular primal-dual KKT matrix under consideration, then the *LBL^T* factorization defines a permutation P , a block-diagonal B , and a unit-lower-triangular L such that

$$(4.3) \quad P^T K P = L B L^T, \quad \text{where } B = \text{diag}(B_1, B_2, \dots, B_s).$$

Each B_j is either one-by-one, or is two-by-two and nonsingular. The permutation P incorporates certain symmetric interchanges that are needed to preserve numerical stability.

At the start of the ℓ th stage ($\ell \geq 1$) of the *LBL^T* factorization of K we have

$$(4.4) \quad K^{(\ell)} = \begin{pmatrix} K_{ij}^{(\ell)} \end{pmatrix} = P_\ell^T K P_\ell - \sum_{i=1}^{\ell-1} L_i B_i L_i^T = \begin{pmatrix} 0 & 0 \\ 0 & K_\ell \end{pmatrix},$$

where P_ℓ is a permutation and L_i is either an $n \times 1$ or $n \times 2$ matrix consisting of the column(s) of L computed at the i th stage. The matrix K_ℓ is called the *Schur complement* and represents the part of K remaining to be factorized. At the ℓ th stage, a one-by-one or symmetric two-by-two submatrix of K_ℓ is selected as the next pivot B_ℓ . The pivot rows and columns are brought to the leading position of K_ℓ using a symmetric interchange, which must be applied to $P_\ell^T K P_\ell$ and the rows of each L_i . The one-by-one or two-by-two pivot is then used to eliminate one or two rows and columns from the permuted Schur complement. The ℓ th stage is completed by the computation of L_ℓ from B_ℓ^{-1} and the pivot columns of $K^{(\ell)}$ (for further details, see Bunch [3]).

Clearly, the permutation P depends on the sequence of pivots. Various pivoting strategies have been proposed that define a numerically stable factorization (see, e.g.,

Bunch, Kaufman, and Parlett [5]; Higham [26]). Here we consider pivot strategies that additionally ensure that B has a specific inertia.

If K is the primal-dual matrix (4.1), potential pivots in the Schur complement can be labeled to reflect the position of the pivot in the original matrix. For example, a diagonal of the Schur complement will be a “ D pivot” if it is in the position occupied by an element of $-D$ in the original matrix. Similarly, a two-by-two pivot with diagonal and off-diagonal positions occupied by elements from H and D will be an “ HD pivot.” With this labeling, all one-by-one pivots are either H or D pivots, and two-by-two pivots are HH , DD , or HD (DH) pivots.

The key to the calculation of suitable search directions is the use of a sequence of *inertia-controlling pivots*. The resulting factorization generalizes the symmetric indefinite factorization proposed by Forsgren and Murray [17] for linear equality-constrained minimization.

The inertia-controlling factorization consists of two phases. The first phase continues until all rows and columns associated with the D block of K have been eliminated (i.e., until no D pivots or HD pivots remain in the Schur complement). During this first phase, only certain types of pivots are allowed (see below). When the first phase is complete, the pivot choice is restricted only by numerical considerations.

In order to describe the first phase it is necessary to further distinguish between pivot types. Let a superscript “+” or “−” indicate the sign of an eigenvalue of a potential pivot. For example, H^+ denotes a positive one-by-one H pivot; D^- denotes a negative one-by-one D pivot; and HD^{-+} denotes an HD pivot with one positive and one negative eigenvalue. The main feature of the first phase of the factorization is that only H^+ , D^- , HH^{++} , DD^{--} , and HD^{-+} pivots are allowed.

Consider the partition of K given by

$$\begin{pmatrix} H_{11} & H_{12} & J_1^T \\ H_{21} & H_{22} & J_2^T \\ J_1 & J_2 & -D \end{pmatrix},$$

and assume that all pivots from the first phase come from rows and columns associated with H_{11} , J_1 , and $-D$. Without loss of generality, we assume the permutation P to be such that

$$P^T K P = \begin{pmatrix} H_{11} & J_1^T & H_{21} \\ J_1 & -D & J_2 \\ H_{21} & J_2^T & H_{22} \end{pmatrix},$$

i.e., the order of the pivots *within* the first phase is ignored. The matrix $P^T K P$ is partitioned as

$$P^T K P = \begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix},$$

where

$$K_{11} = \begin{pmatrix} H_{11} & J_1^T \\ J_1 & -D \end{pmatrix}, \quad K_{21} = K_{12}^T = (H_{21} \quad J_2^T), \quad \text{and} \quad K_{22} = H_{22}.$$

The sequence of pivots used during the first phase fixes K_{11} as an $n_1 \times m$ matrix for some n_1 ($0 \leq n_1 \leq n$) and thereby identifies the blocks K_{11} , K_{12} , and K_{22} (the value

of n_1 will depend upon the relative number of one-by-one and two-by-two pivots). The restricted pivot choice ensures that $\text{In}(K_{11}) = (n_1, m, 0)$. Moreover, on completion of the first phase, factors L_{11} and B_{11} have been computed such that

$$K_{11} = L_{11}B_{11}L_{11}^T \quad \text{and} \quad L_{21} = K_{21}L_{11}^{-T}B_{11}^{-1}.$$

During the second phase, the pivot choice is unrestricted, which implies that the remaining Schur complement $K_{22} - K_{21}K_{11}^{-1}K_{12}$ can be factored using a standard LBL^T -factorization. The only restriction is that the choice of pivots gives a factorization for which $\|L\|$ is bounded.

On completion of the second phase we have

$$K_{22} - K_{21}K_{11}^{-1}K_{12} = L_{22}B_{22}L_{22}^T.$$

This provides the complete factorization

$$\begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix} = \begin{pmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{pmatrix} \begin{pmatrix} B_{11} & 0 \\ 0 & B_{22} \end{pmatrix} \begin{pmatrix} L_{11}^T & L_{21}^T \\ 0 & L_{22}^T \end{pmatrix}.$$

4.2. Calculation of the descent direction. Given the inertia-controlling LBL^T -factorization of K , the descent direction is now computed from

$$(4.5) \quad \begin{pmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{pmatrix} \begin{pmatrix} B_{11} & 0 \\ 0 & \bar{B}_{22} \end{pmatrix} \begin{pmatrix} L_{11}^T & L_{21}^T \\ 0 & L_{22}^T \end{pmatrix} \tilde{s} = -P^T \begin{pmatrix} g - J^T \lambda \\ D(\lambda - \pi) \end{pmatrix},$$

with

$$\begin{pmatrix} s_x \\ -s_\lambda \end{pmatrix} = P\tilde{s},$$

where $\bar{B}_{22}$ is a sufficiently positive-definite modification of B_{22} . In particular, if B_{22} is sufficiently positive definite, then $\bar{B}_{22} = B_{22}$. It is relatively straightforward to modify B_{22} since the diagonal blocks are of size one-by-one or two-by-two and the eigenvalues of each block can be found explicitly and modified if necessary. A crucial feature of the factorization is that K_{22} only includes rows and columns from H . This ensures that any modification of B_{22} affects only parts of H in K . Furthermore, (4.5) can be written as

$$(4.6) \quad \begin{pmatrix} \bar{H} & J^T \\ J & -D \end{pmatrix} \begin{pmatrix} s_x \\ -s_\lambda \end{pmatrix} = - \begin{pmatrix} g - J^T \lambda \\ D(\lambda - \pi) \end{pmatrix},$$

where $\bar{H}$ is a symmetric modification of H such that $\bar{H} + J^T D^{-1} J$ is sufficiently positive definite. Whenever $H + J^T D^{-1} J$ is sufficiently positive definite, no modification takes place and the descent direction s is the primal-dual direction computed from (2.10). This is analogous to the method of Forsgren and Murray [17].

Alternatively, we may use Theorem 3.2 to establish that (4.6) is equivalent to

$$\bar{S}s = -\nabla M^{\mu,\nu}(x, \lambda),$$

where

$$(4.7) \quad \bar{S} = \begin{pmatrix} \bar{H} + (1 + \nu)J^T D^{-1} J & \nu J^T \\ \nu J & \nu D \end{pmatrix}.$$

We first establish descent properties of s with respect to the merit function (3.1).

LEMMA 4.2. *Let s satisfy (4.6), where $\bar{H}$ is symmetric, D is symmetric and positive definite, and $\bar{H} + J^T D^{-1} J$ is positive definite. Then*

$$s^T \nabla M^{\mu, \nu}(x, \lambda) = -s_x^T (\bar{H} + J^T D^{-1} J) s_x - \nu(\lambda - \pi)^T D(\lambda - \pi),$$

where $\nabla M^{\mu, \nu}(x, \lambda)$ is defined by (3.2a). Moreover, $s_x = 0$ if and only if $g - J^T \pi = 0$. Finally, if $s_x = 0$, then $s_\lambda = \pi - \lambda$.

Proof. Straightforward substitution gives the first result. Elimination of s_λ in (4.6) gives

$$(\bar{H} + J^T D^{-1} J) s_x = -(g - J^T \pi).$$

Hence, since $\bar{H} + J^T D^{-1} J$ is positive definite, we have $s_x = 0$ if and only if $g - J^T \pi = 0$. Finally, if $s_x = 0$, then $g = J^T \pi$, and s_λ is given by

$$\begin{pmatrix} -J^T \\ D \end{pmatrix} s_\lambda = - \begin{pmatrix} g - J^T \pi \\ D(\lambda - \pi) \end{pmatrix} = \begin{pmatrix} -J^T \\ D \end{pmatrix} (\pi - \lambda),$$

with unique solution $s_\lambda = \pi - \lambda$. $\square$

The positive definiteness of $\bar{S}$ ensures that $s^T \nabla M^{\mu, \nu}(x, \lambda) < 0$ unless $s = 0$ and $\nabla M^{\mu, \nu}(x, \lambda) = 0$. The additional information provided by Lemma 4.2 is that if $s_x = 0$, then $g - J^T \pi = 0$ and $s_\lambda = \pi - \lambda$. This means that if $x = x(\mu)$, then one further Newton step gives the λ variables as $\lambda(\mu)$.

Some further assumptions are required in order to make s a sufficient descent direction in the sense of (3.4a). This is summarized in the following theorem.

THEOREM 4.3. *Assume that the sequence of directions $\{s_k\}_{k=0}^\infty$ satisfies*

$$\begin{pmatrix} \bar{H}_k & J_k^T \\ J_k & -D_k \end{pmatrix} \begin{pmatrix} s_{x,k} \\ -s_{\lambda,k} \end{pmatrix} = - \begin{pmatrix} g_k - J_k^T \lambda_k \\ D_k(\lambda_k - \pi_k) \end{pmatrix},$$

where $\bar{H}_k + J_k^T D_k^{-1} J_k \succ 0$ and $D_k \succ 0$ for all k . Then, $s_k^T \nabla M^{\mu, \nu}(x_k, \lambda_k) < 0$ unless $s_k = 0$. Furthermore, assume that $\limsup_{k \rightarrow \infty} \|\bar{H}_k\| < \infty$, $\limsup_{k \rightarrow \infty} \|J_k\| < \infty$, $\liminf_{k \rightarrow \infty} \eta_{\min}(D_k) > 0$, $\limsup_{k \rightarrow \infty} \eta_{\max}(D_k) < \infty$, and $\liminf_{k \rightarrow \infty} \eta_{\min}(\bar{H}_k + J_k^T D_k^{-1} J_k) > 0$. Then, if

$$\lim_{k \rightarrow \infty} s_k^T \nabla M^{\mu, \nu}(x_k, \lambda_k) = 0,$$

it holds that

$$\lim_{k \rightarrow \infty} s_k = 0 \quad \text{and} \quad \lim_{k \rightarrow \infty} \nabla M^{\mu, \nu}(x_k, \lambda_k) = 0.$$

Proof. Theorem 3.2 gives $\bar{S}_k s_k = -\nabla M^{\mu, \nu}(x_k, \lambda_k)$, where

$$\bar{S}_k = \begin{pmatrix} \bar{H}_k + (1 + \nu) J_k^T D_k^{-1} J_k & \nu J_k^T \\ \nu J_k & \nu D_k \end{pmatrix}.$$

If $\bar{H}_k + J_k^T D_k^{-1} J_k \succ 0$, $D_k \succ 0$, and $\nu > 0$, Lemma 4.1 shows that $\bar{S}_k \succ 0$. Hence, the identity $s_k^T \nabla M^{\mu, \nu}(x_k, \lambda_k) = -s_k^T \bar{S}_k s_k$ ensures that $s_k^T \nabla M^{\mu, \nu}(x_k, \lambda_k) < 0$ unless $s_k = 0$.

To prove the second part of the theorem, $\bar{S}_k$ is written as

$$(4.8) \quad \bar{S}_k = \begin{pmatrix} I & J_k^T D_k^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} \bar{H}_k + J_k^T D_k^{-1} J_k & 0 \\ 0 & \nu D_k \end{pmatrix} \begin{pmatrix} I & 0 \\ D_k^{-1} J_k & I \end{pmatrix}.$$

It follows from (4.8) and the boundedness assumptions that $\liminf_{k \rightarrow \infty} \eta_{\min}(\bar{S}_k) > 0$ and $\limsup_{k \rightarrow \infty} \eta_{\max}(\bar{S}_k) < \infty$ (see Lemma 3.1 in Forsgren and Murray [17]). The proof is completed by combining these results with the modified Newton equation $\bar{S}_k s_k = -\nabla M^{\mu, \nu}(x_k, \lambda_k)$. $\square$

4.3. Calculation of the direction of negative curvature. Given the inertia-controlling LBL^T -factorization of K , if B_{22} has at least one negative eigenvalue, the direction of negative curvature is computed from

$$(4.9) \quad \begin{pmatrix} L_{11}^T & L_{21}^T \\ 0 & L_{22}^T \end{pmatrix} \tilde{d} = \pm \sigma \begin{pmatrix} 0 \\ u \end{pmatrix},$$

with

$$\begin{pmatrix} d_x \\ -d_\lambda \end{pmatrix} = P \tilde{d},$$

where $\sigma = \sqrt{-\eta_{\min}(B_{22})}$ and u is an associated eigenvector of unit length. (Note that σ and u are easily obtained since B_{22} is symmetric block-diagonal with diagonal blocks of dimension one-by-one or two-by-two.) The sign of d is chosen such that $d^T \nabla M^{\mu, \nu}(x, \lambda) \leq 0$. Since $K_{22} = H_{22}$ it follows from (4.9) that $Jd_x + Dd_\lambda = 0$. Also, from the definition of d it follows that $d^T Sd = -\eta_{\min}(B_{22})^2$.

The following lemma establishes the properties of the curvature along a vector d such that $Jd_x + Dd_\lambda = 0$.

LEMMA 4.4. *If $Jd_x + Dd_\lambda = 0$ and D is symmetric and nonsingular, then*

$$\begin{aligned} d_x^T (H + J^T D^{-1} J) d_x &= \begin{pmatrix} d_x^T & -d_\lambda^T \end{pmatrix} \begin{pmatrix} H & J^T \\ J & -D \end{pmatrix} \begin{pmatrix} d_x \\ -d_\lambda \end{pmatrix} \\ &= \begin{pmatrix} d_x^T & d_\lambda^T \end{pmatrix} \begin{pmatrix} H + (1 + \nu) J^T D^{-1} J & \nu J^T \\ \nu J & \nu D \end{pmatrix} \begin{pmatrix} d_x \\ d_\lambda \end{pmatrix}. \end{aligned}$$

Proof. Straightforward substitution using $Jd_x + Dd_\lambda = 0$ gives the result. $\square$

Finally, we give a sufficient curvature result for a sequence of iterates.

THEOREM 4.5. *Assume that each member of the sequence $\{d_k\}_{k=0}^\infty$ is computed using the system (4.9). Assume that this system is associated with the LBL^T -factors of a sequence of B_k -matrices such that for each k , B_k has more than m negative eigenvalues. Furthermore, assume that the factorization is inertia controlling, with $\limsup_{k \rightarrow \infty} \|L_k\| < \infty$. Then, if*

$$\lim_{k \rightarrow \infty} d_k^T S_k d_k = 0,$$

it holds that

$$\lim_{k \rightarrow \infty} d_k = 0 \quad \text{and} \quad \lim_{k \rightarrow \infty} \eta_{\min}(S_k) = 0.$$

Proof. It follows from (4.9) that $d_k^T S_k d_k = -\eta_{\min}(B_{22,k})^2$. Hence, if d_k satisfies $\lim_{k \rightarrow \infty} d_k^T S_k d_k = 0$, then it must be the case that $\lim_{k \rightarrow \infty} \eta_{\min}(B_{22,k}) = 0$. Now, the definition (4.9) of d_k and the assumption that $\limsup_{k \rightarrow \infty} \|L_k\| < \infty$ ensure that $\lim_{k \rightarrow \infty} d_k = 0$. Since $\text{In}(K_{11,k}) = (n_{1,k}, m, 0)$, we may view $K_{22,k} - K_{21,k} K_{11,k}^{-1} K_{12,k}$ as being formed by first eliminating $-D_k$, giving the m negative eigenvalues. The Schur complement obtained this way is $H_k + J_k^T D_k^{-1} J_k$. Hence, elimination of the remaining $n_{1,k} - m$ pivots of $K_{11,k}$ must have been done by positive-definite pivots. Since K_k has more than m negative eigenvalues, we must have $\eta_{\min}(H_k + J_k^T D_k^{-1} J_k) < 0$. We conclude that

$$(4.10) \quad \eta_{\min}(K_{22,k} - K_{21,k} K_{11,k}^{-1} K_{12,k}) \leq \eta_{\min}(H_k + J_k^T D_k^{-1} J_k) < 0;$$

see Lemma 2.1 in Forsgren, Gill, and Murray [15]. By assumption, the factorization guarantees that $\limsup_{k \rightarrow \infty} \|L_k\| < \infty$. Since L_k is unit-lower-triangular, this implies that $\limsup_{k \rightarrow \infty} \|L_{22,k}\| < \infty$ and $\limsup_{k \rightarrow \infty} \|L_{22,k}^{-1}\| < \infty$. Hence, if $\lim_{k \rightarrow \infty} \eta_{\min}(B_{22,k}) = 0$, we must have

$$(4.11) \quad \lim_{k \rightarrow \infty} \eta_{\min}(K_{22,k} - K_{21,k} K_{11,k}^{-1} K_{12,k}) = 0;$$

see Lemma 3.1 in Forsgren and Murray [17]. A combination of (4.10), (4.11), and Lemma 4.1 gives $\lim_{k \rightarrow \infty} \eta_{\min}(S_k) = 0$, as required. $\square$

4.4. Properties of the search direction. The sufficiency properties of Theorems 4.3 and 4.5 for the descent direction and direction of negative curvature require L and D to be bounded and D to be bounded away from a singular matrix. In addition, we have to assume that the iterates remain in a compact set for convergence to be assured. In theory, this can be handled by adding lower and upper bounds on the variables since the algorithm stays feasible with respect to all inequalities. However, this is not completely satisfactory. In this paper, the discussion on convergence is limited to the results of the previous section. A formal convergence proof will be considered in a future paper.

At any iteration, an alternative to performing a line search with respect to x and λ is to reset λ to $\pi^\mu(x)$ and perform the line search with respect to x only. In this case, the directions defined by the primal-dual equations (2.8) and the primal equations (2.9) are identical. This strategy allows the use of a convergence proof for the pure primal penalty-barrier method. Unfortunately, we know of no published proof for the nonconvex case that gives limit points that satisfy the second-order necessary optimality conditions. For line-search barrier-penalty-type methods, Benchakroun, Dussault, and Mansouri [2], Gould [23], and Dussault [11] discuss convergence to a first-order point. In a trust-region setting, Byrd, Gilbert, and Nocedal [7] propose a method that converges to a first-order point.

The properties required of D for the analysis are typical. For an equality constraint, $d_i^\mu(x, \lambda)$ is fixed at μ , which is trivially bounded away from zero for μ fixed. However, this lower bound obviously deteriorates as μ gets small. For inequality constraints, $d_i^\mu(x, \lambda)$ is $c_i(x)/\lambda$, and on the trajectory we have $d_i^\mu(x(\mu), \lambda(\mu)) = c_i(x(\mu))^2/\mu$. Again, for a fixed μ , we expect $d_i^\mu(x(\mu), \lambda(\mu))$ to be both bounded away from zero and bounded from above. However, further research is required to show this rigorously and to incorporate a suitable scheme for changing μ .

5. Discussion and further research. We have demonstrated how the primal-dual search directions (2.8) can be interpreted as modified Newton directions for minimizing an augmented penalty-barrier merit function (3.1). The main benefits of this

approach are twofold: (i) the equations are well conditioned in the limit when strict complementarity holds; and (ii) the merit function is minimized with respect to both x and λ . The inertia-controlling factorization provides descent directions for $M^{\mu,\nu}(x, \lambda)$ and directions of negative curvature for the approximate Hessian $S^{\mu,\nu}(x, \lambda)$. We observe that the ill-conditioning has been removed from the equations determining the search direction and is now only present in the merit function.

If the proposed method is to work well, suitable termination criteria must be determined for the outer iterations. The convergence analysis must be extended to an algorithm that includes a strategy for reducing μ . In the limit, this algorithm should require only one inner iteration for each outer iteration. However, although preliminary numerical results have been encouraging, it is not known at this stage whether or not the merit function can induce the “Maratos effect,” whereby the merit function does not accept the unit step as μ is decreased.

Finally, we note that the proposed method is applicable to sparse problems. The key is the calculation of an efficient sparse inertia-controlling factorization of K (4.1). Up to now, our experiments have been limited to the use of a dense factorization. (Implementations of other sparse inertia-controlling factorizations are discussed by Arioli et al. [1] and Gould [24, 25] in the context of linear equality constrained optimization.)

Other interesting variations of the proposed method remain to be investigated. For example, if (2.5b) is replaced by $\mu + c_i(x)/\lambda_i$ for $i \in \mathcal{E}$, then the Jacobian of the resulting system is well conditioned in the limit if strict complementarity holds. An advantage of this formulation is that the Jacobian is independent of μ . However, we have not yet been able to associate these equations with a suitable merit function. Another possibility is to use different values of μ and ν for each constraint. A zero value of μ can be assigned to any linear equality constraint once it is satisfied. This would give a method equivalent to that of Forsgren and Murray [17]. If only equality constraints are present, a zero value of μ gives the standard sequential quadratic programming search direction. However, if a constraint is not satisfied exactly, a zero value of μ is not treated by the analysis of this paper. The introduction of shifts on the constraints may be helpful in this case (see Powell [39]).

Acknowledgments. We are grateful to Annick Sartenaer, Philippe Toint, and an anonymous referee for their careful reading of the manuscript and several suggestions that improved the presentation.

REFERENCES

- [1] M. ARIOLI, T. F. CHAN, I. S. DUFF, N. I. M. GOULD, AND J. K. REID, *Computing a Search Direction for Large-Scale Linearly Constrained Nonlinear Optimization Calculations*, Tech. Rep. RAL-93-066, Rutherford Appleton Laboratory, Chilton, England, 1993.
- [2] A. BENCHAKROUN, J. DUSSAULT, AND A. MANSOURI, *A two parameter mixed interior-exterior penalty algorithm*, ZOR—Math. Methods Oper. Res., 41 (1995), pp. 25–55.
- [3] J. R. BUNCH, *Partial pivoting strategies for symmetric matrices*, SIAM J. Numer. Anal., 11 (1974), pp. 521–528.
- [4] J. R. BUNCH AND L. KAUFMAN, *Some stable methods for calculating inertia and solving symmetric linear systems*, Math. Comp., 31 (1977), pp. 163–179.
- [5] J. R. BUNCH, L. KAUFMAN, AND B. N. PARLETT, *Decomposition of a symmetric matrix*, Numer. Math., 27 (1976), pp. 95–109.
- [6] J. R. BUNCH AND B. N. PARLETT, *Direct methods for solving symmetric indefinite systems of linear equations*, SIAM J. Numer. Anal., 8 (1971), pp. 639–655.

- [7] R. H. BYRD, J. C. GILBERT, AND J. NOCEDAL, *A Trust Region Method Based on Interior Point Techniques for Nonlinear Programming*, Tech. Rep. OTC-96/2, Optimization Technology Center, Northwestern University, Evanston, IL, 1996.
- [8] A. R. CONN, N. I. M. GOULD, AND P. L. TOINT, *A note on using alternative second-order models for the subproblems arising in barrier function methods for minimization*, Numer. Math., 68 (1994), pp. 17–33.
- [9] R. W. COTTLE, *Manifestations of the Schur complement*, Linear Algebra Appl., 8 (1974), pp. 189–211.
- [10] J. E. DENNIS, JR. AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1983.
- [11] J.-P. DUSSAULT, *Numerical stability and efficiency of penalty algorithms*, SIAM J. Numer. Anal., 32 (1995), pp. 296–317.
- [12] A. S. EL-BAKRY, R. A. TAPIA, Y. ZHANG, AND T. TSUCHIYA, *On the Formulation and Theory of the Newton Interior Point Method for Nonlinear Programming*, Tech. Report 92-40, Department of Computational and Applied Mathematics, Rice University, Houston, TX, 1992. Revised April 1995.
- [13] A. V. FIACCO AND G. P. MCCORMICK, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley and Sons, New York, London, Sydney, Toronto, 1968.
- [14] R. FLETCHER, *Factorizing symmetric indefinite matrices*, Linear Algebra Appl., 14 (1976), pp. 257–272.
- [15] A. FORSGREN, P. E. GILL, AND W. MURRAY, *Computing modified Newton directions using a partial Cholesky factorization*, SIAM J. Sci. Comput., 16 (1995), pp. 139–150.
- [16] A. FORSGREN, P. E. GILL, AND J. R. SHINNERL, *Stability of symmetric ill-conditioned systems arising in interior methods for constrained optimization*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 187–211.
- [17] A. FORSGREN AND W. MURRAY, *Newton methods for large-scale linear equality-constrained minimization*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 560–587.
- [18] P. E. GILL, W. MURRAY, D. B. PONCELEÓN, AND M. A. SAUNDERS, *Primal-dual methods for linear programming*, Math. Programming, 70 (1995), pp. 251–277.
- [19] P. E. GILL, W. MURRAY, AND M. H. WRIGHT, *Practical Optimization*, Academic Press, London, New York, 1981.
- [20] D. GOLDFARB, *Curvilinear path steplength algorithms for minimization which use directions of negative curvature*, Math. Programming, 18 (1980), pp. 31–40.
- [21] C. C. GONZAGA, *Path-following methods for linear programming*, SIAM Rev., 34 (1992), pp. 167–224.
- [22] N. I. M. GOULD, *On the accurate determination of search directions for simple differentiable penalty functions*, IMA J. Numer. Anal., 6 (1986), pp. 357–372.
- [23] N. I. M. GOULD, *On the convergence of a sequential penalty function method for constrained minimization*, SIAM J. Numer. Anal., 26 (1989), pp. 107–128.
- [24] N. I. M. GOULD, *Constructing Appropriate Models for Large-Scale, Linearly Constrained, Nonconvex, Nonlinear Optimization Algorithms*, Tech. Rep. RAL-TR-95-037, Rutherford Appleton Laboratory, Chilton, England, 1995.
- [25] N. I. M. GOULD, *Constructing appropriate models for large-scale, linearly-constrained, non-convex, nonlinear optimization algorithms*, SIAM J. Optim., submitted.
- [26] N. J. HIGHAM, *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, PA, 1996.
- [27] S. KANIEL AND A. DAX, *A modified Newton's method for unconstrained minimization*, SIAM J. Numer. Anal., 16 (1979), pp. 324–331.
- [28] M. KOJIMA, S. MIZUNO, AND A. YOSHISE, *A primal-dual interior point algorithm for linear programming*, in Progress in Mathematical Programming: Interior Point and Related Methods, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 29–47.
- [29] I. J. LUSTIG, R. E. MARSTEN, AND D. F. SHANNO, *Computational experience with a primal-dual interior point method for linear programming*, Linear Algebra Appl., 152 (1991), pp. 191–222.
- [30] G. P. MCCORMICK, *A modification of Armijo's step-size rule for negative curvature*, Math. Programming, 13 (1977), pp. 111–115.
- [31] N. MEGIDDO, *Pathways to the optimal set in linear programming*, in Progress in Mathematical Programming: Interior Point and Related Methods, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 131–158.
- [32] S. MEHROTRA, *On the implementation of a primal-dual interior point method*, SIAM J. Optim., 2 (1992), pp. 575–601.

- [33] R. D. C. MONTEIRO AND I. ADLER, *Interior path-following primal-dual algorithms. Part II: Convex quadratic programming*, Math. Programming, 44 (1989), pp. 43–66.
- [34] J. J. MORÉ AND D. C. SORESENSEN, *On the use of directions of negative curvature in a modified Newton method*, Math. Programming, 16 (1979), pp. 1–20.
- [35] H. MUKAI AND E. POLAK, *A second-order method for unconstrained optimization*, J. Optim. Theory Appl., 26 (1978), pp. 501–513.
- [36] W. MURRAY, *Analytical expressions for the eigenvalues and eigenvectors of the Hessian matrices of barrier and penalty functions*, J. Optim. Theory Appl., 7 (1971), pp. 189–196.
- [37] W. MURRAY AND M. H. WRIGHT, *Line search procedures for the logarithmic barrier function*, SIAM J. Optim., 4 (1994), pp. 229–246.
- [38] D. B. PONCELEÓN, *Barrier Methods for Large-Scale Quadratic Programming*, Ph.D. thesis, Report SOL 91-2, Department of Operations Research, Stanford University, Stanford, CA, 1991.
- [39] M. J. D. POWELL, *A method for nonlinear constraints in minimization problems*, in Optimization, R. Fletcher, ed., Academic Press, New York, 1969, pp. 283–298.
- [40] K. SVANBERG, *Method of moving asymptotes—a new method for structural optimization*, Internat. J. Numer. Methods Engrg., 24 (1987), pp. 359–373.
- [41] M. H. WRIGHT, *Some properties of the Hessian of the logarithmic barrier function*, Math. Programming, 67 (1994), pp. 265–295.
- [42] M. H. WRIGHT, *Why a pure primal Newton barrier step may be infeasible*, SIAM J. Optim., 5 (1995), pp. 1–12.
- [43] S. J. WRIGHT, *Stability of linear equations solvers in interior-point methods*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 1287–1307.
- [44] S. WRIGHT, *Stability of augmented system factorizations in interior-point methods*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 191–222.
- [45] Y. YE, *Line Searches in Potential Reduction Algorithm for Linear Programming*, manuscript, 1989.