COMPUTING THE INTEGER PROGRAMMING GAP

SERKAN HOŞTEN, BERND STURMFELS*

We determine the maximal gap between the optimal values of an integer program and its linear programming relaxation, where the matrix and cost function are fixed but the right hand side is unspecified. Our formula involves irreducible decomposition of monomial ideals. The gap can be computed in polynomial time when the dimension is fixed.

1. Introduction

We consider the general integer programming problem in standard form,

(1)   Minimize $c \cdot u$ subject to $A \cdot u = b$, $u \geq 0$, u integral.

Here A is a fixed $d \times n$ integer matrix, $b \in \mathbb{Z}^d$ and $c \in \mathbb{Q}^n$. Its linear programming relaxation is obtained by dropping the integrality constraints,

(2)   Minimize $c \cdot u$ subject to $A \cdot u = b$, $u \geq 0$.

Suppose that the integer program (1) is feasible and bounded. Then the linear program (2) is feasible and bounded as well, and the optimal value of (1) is greater than or equal to the optimal value of (2). We define the nonnegative rational number gap(A,c) to be the maximum difference of the two optimal values as b ranges over vectors such that (1) is feasible and bounded. It follows from known results [17, Theorem 17.2] that this maximum is bounded.

Our main aim in this paper is to provide an exact formula for $\text{gap}(A,c)$. We express our results using the language of Gröbner bases, as in [4, Chapter 8], [11], [16, § 4.4]. A nonnegative integer vector $u = (u_1,\ldots,u_n) \in \mathbb{N}^n$ is called *non-optimal* if it is not an optimal solution of (1) with $b = A \cdot u$. We represent each non-optimal vector u by a monomial $x^u = x_1^{u_1} x_2^{u_2} \cdots x_n^{u_n}$, and we consider the ideal $M(A,c)$ generated by these monomials in the polynomial ring $\mathbb{R}[x_1,\ldots,x_n]$. If c is generic then $M(A,c)$ is an initial ideal of the toric ideal of A; see [18]. In this case, $M(A,c)$ can be read off from the corresponding Gröbner basis of the toric ideal. If c is not generic then we can compute $M(A,c)$ using [16, Algorithm 4.4.2]. A monomial ideal I in $\mathbb{R}[x_1,\ldots,x_n]$ is called *irreducible* if it is generated by powers of the variables:

$$I = \left\langle x_{i_1}^{u_{i_1}+1}, x_{i_2}^{u_{i_2}+1}, \ldots, x_{i_r}^{u_{i_r}+1} \right\rangle.$$

Every monomial ideal M in $\mathbb{R}[x_1,\ldots,x_n]$ can be written uniquely as an irredundant intersection of finitely many irreducible monomial ideals, which are called the *irreducible components* of M. Suppose that I is an irreducible component of $M(A,c)$. We define the *gap value* of I with respect to A and c to be the optimal objective function value of the auxiliary linear program

(3)
$$\begin{aligned} \text{Maximize} \quad & u_{i_1} c_{i_1} + u_{i_2} c_{i_2} + \cdots + u_{i_r} c_{i_r} - c \cdot v \\ \text{subject to} \quad & A \cdot v = u_{i_1} \cdot A_{i_1} + u_{i_2} \cdot A_{i_2} + \cdots + u_{i_r} \cdot A_{i_r}, \\ \text{and} \quad & v_{i_1}, \ldots, v_{i_r} \geq 0. \end{aligned}$$

Here $A_1, A_2, \ldots, A_n$ are the column vectors of the matrix A.

Theorem 1.1. *The integer programming gap, $\text{gap}(A,c)$, equals the maximum gap value of any irreducible component I of the monomial ideal $M(A,c)$.*

We remark that $\text{gap}(A,c)$ is zero if and only if the monomial ideal $M(A,c)$ is generated by squarefree monomials $x_{j_1} x_{j_2} \cdots x_{j_r}$. This happens, for instance, if the coefficient matrix A is totally unimodular.

This paper is organized as follows. In Section 2 we rephrase our problem in the more general setting of lattice programs, and we prove Theorem 1.1 in this context. In Section 3 we apply the work of Barvinok and Woods [3] on short rational generating functions to derive the following complexity result.

Theorem 1.2. *For fixed d and n, the integer programming gap, $\text{gap}(A,c)$, can be computed in polynomial time in the binary encoding of A and c.*

We conjecture that the same result holds if n and d are allowed to vary but their difference $n-d$ is fixed. This would require techniques beyond [3].

Section 4 concerns applications to the statistical theory of multidimensional contingency tables. Here we are interested in the integer programming gap of certain higher-dimensional transportation problems. These play an important role in data security. For the statistical background see [5], [8].

In Section 5 we vary the cost vector c, and we prove that the function

$$(4) \qquad \mathrm{gap}_A \ : \ \mathbb{R}^n \to \mathbb{R}, \ c \mapsto \mathrm{gap}(A, c)$$

is a piecewise-linear function on $\mathbb{R}^n$. We show that gap_A is linear on the cones of a fan which refines the familiar *Gröbner fan* (cf. [13] and [19]).

We close the introduction with a small example. Let (1) be the problem of *making change* using pennies, nickels, dimes and quarters, where the number of coins is fixed, and nickels and quarters are used most sparingly. In symbols,

$$d = 2, \ n = 4, \ A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 5 & 10 & 25 \end{bmatrix}, \ c = \begin{bmatrix} 0 & 1 & 0 & 1 \end{bmatrix}.$$

Using notation as in [9, § I.5], this problem is solved by the Gröbner basis

$$\mathcal{G} = \{\underline{n^3 q} - d^4, \underline{n^6} - p^5 q, \underline{n^3 d^4} - p^5 q^2, \underline{p^5 q^3} - d^8\}.$$

The four underlined leading monomials generate the ideal

$$(5) \qquad M(A, c) = \langle n^3, p^5 \rangle \cap \langle n^3, q^3 \rangle \cap \langle n^6, d^4, q \rangle.$$

The gap values of the three irreducible components are $76/15$, 4 and 5. Hence

$$\mathrm{gap}(A, c) = 76/15 = 5.0666666\ldots$$

This gap is attained when expressing one dollar and 14 cents with ten coins. The optimal solutions are $(4, 2, 0, 4)$ and $(0, 0, 136/15, 14/15)$ respectively.

2. The gap theorem for lattice programs

Lattice programs are defined as follows. Let $\mathcal{L}$ be a fixed lattice of rank m in $\mathbb{Z}^n$. Then $\mathcal{L}_\mathbb{R} = \mathcal{L} \otimes_\mathbb{Z} \mathbb{R}$ is an m-dimensional vector space in $\mathbb{R}^n$. We also fix a cost vector $c \in \mathbb{Q}^n$. Now, for every $z \in \mathbb{N}^n$, we get a *lattice program*

$$(6) \qquad \text{Minimize} \quad c \cdot v \quad \text{subject to} \quad v \equiv z \bmod \mathcal{L}, \quad v \in \mathbb{N}^n,$$

and its *linear relaxation*

$$(7) \qquad \text{Minimize} \quad c \cdot v \quad \text{subject to} \quad v \equiv z \bmod \mathcal{L}_\mathbb{R}, \quad v \geq 0.$$

We define the *lattice programming gap* $\mathrm{gap}(\mathcal{L}, c)$ to be maximum of the differences between the optimal values of (6) and (7) as z runs over $\mathbb{N}^n$.

Remark. The integer programs (1) are lattice programs with $\mathcal{L} = \ker(A) \cap \mathbb{Z}^n$ and $b = A \cdot z$. The linear programming relaxations (2) correspond to the linear relaxations (7) of these lattice programs. Note that we have $m = n - d$.

Let $M(\mathcal{L}, c)$ be the ideal generated by all monomials x^u where $u \in \mathbb{N}^n$ is non-optimal for (6). Any irreducible component of $M(\mathcal{L}, c)$ has the form

$$I(u, \tau) = \left\langle x_{i_1}^{u_{i_1}+1}, x_{i_2}^{u_{i_2}+1}, \ldots, x_{i_r}^{u_{i_r}+1} \right\rangle,$$

where $\tau = \{i_1, \ldots, i_r\}$ and $u \in \mathbb{N}^n$ with $u_j = 0$ for $j \notin \tau$. The *gap value* of $I(u, \tau)$ with respect to $\mathcal{L}$ and c is the optimal objective function value of

$$
\begin{array}{lll}
(8) & \text{Maximize} & u_{i_1} c_{i_1} + u_{i_2} c_{i_2} + \cdots + u_{i_r} c_{i_r} - c \cdot v \\
& \text{subject to} & v \equiv u \bmod \mathcal{L}_{\mathbb{R}} \quad \text{and} \quad v_{i_1}, \ldots, v_{i_r} \geq 0
\end{array}
$$

We restate Theorem 1.1 for the more general setting of lattice programs.

Theorem 2.1. *The lattice programming gap, $\mathrm{gap}(\mathcal{L}, c)$, equals the maximum gap value of any irreducible component $I(u, \tau)$ of the monomial ideal $M(\mathcal{L}, c)$.*

Example 2.2 $(m = n)$**.** Let $\mathcal{L}$ be a finite index sublattice of $\mathbb{Z}^n$ and c a nonnegative vector. Since $\mathcal{L}_{\mathbb{R}} = \mathbb{R}^n$, the objective function value of (7) is always zero, so $\mathrm{gap}(\mathcal{L}, c)$ is the largest objective function value of (6). The finite abelian group $\mathbb{Z}^n / \mathcal{L}$ is in bijection with the set of monomials not in $M(\mathcal{L}, c)$. The irreducible components $I(u, \{1, \ldots, n\})$ are indexed by monomials $x_1^{u_1} x_2^{u_2} \cdots x_n^{u_n}$ which are maximal with respect to divisibility among those not in $M(\mathcal{L}, c)$. The lattice programming gap is the maximum of the corresponding values $c_1 u_1 + c_2 u_2 + \cdots + c_n u_n$. ∎

Example 2.2 is called the *group problem* in the integer programming literature [17, § 24.2]. More generally, every lattice program (6) has a natural family of *group relaxations* which are indexed by subsets τ of $\{1, 2, \ldots, n\}$:

$$(9) \quad \text{Minimize} \quad c \cdot v \quad \text{subject to} \quad v \equiv z \bmod \mathcal{L}, \quad v \in \mathbb{Z}^n, \; v_i \geq 0 \; \forall i \in \tau.$$

If an optimal solution v^* of (9) is a nonnegative vector then v^* is also feasible and optimal for (6). In this case we say that lattice program (6) is *solved by* τ. The minimal collection of required subsets τ is studied in [12]. The following result is well-known in the algebraic theory of integer programming; see [11, § 3]. The pairs (u, τ) in Proposition 2.3 are called *standard pairs*. A combinatorial introduction can be found in [21].

Proposition 2.3. *There is a unique minimal finite set $\mathcal{S}$ of irreducible ideals $I(u,\tau)$ whose intersection is $M(\mathcal{L},c)$ such that every optimal solution v^* to (6) has the form $v^* = u + v'$, with $v_i' = 0$ for $i \in \tau$, for some $I(u,\tau) \in \mathcal{S}$.*

When a lattice program (6) has an optimal solution $v^* = u + v'$ as in Proposition 2.3, we shall say that it is *solved by the standard ideal $I(u,\tau)$*. In this case the group relaxation (9) has the same optimal solution v^*.

Remark. The irreducible components of $M(\mathcal{L},c)$ are a subset of $\mathcal{S}$. In fact, the irreducible components are the minimal elements of $\mathcal{S}$ when the standard ideals are ordered with respect to inclusion. In the special case of Example 2.2, we have $\mathcal{S} = \{I(u,\{1,\ldots,n\}) : x^u \notin M(\mathcal{L},c)\}$.

Lemma 2.4. *Fix $I(u,\tau) \in \mathcal{S}$. The gap value of $I(u,\tau)$ equals the maximum difference between the optimal values of (6) and (7) as z ranges over all vectors in $\mathbb{N}^n$ such that the program (6) is solved by the standard ideal $I(u,\tau)$.*

Proof. Suppose a lattice program (6) is solved by the standard ideal $I(u,\tau)$ and has the optimal solution $x^* = u + x'$ where $x_i' = 0$ for all $i \in \tau$. Let y^* be an optimal solution for the linear relaxation (7). Then the difference between the optimal values is $c \cdot x^* - c \cdot y^*$. Since $y^* - x'$ is a feasible solution for (8), the optimal value of this program is at least $c \cdot u - c \cdot (y^* - x') = c \cdot x^* - c \cdot y^*$.

Hence we only need to find a vector $z \in \mathbb{N}^n$ whose associated lattice program (6) is solved by $I(u,\tau)$, and such that the difference of the optimal values of (6) and (7) is greater than or equal to the optimal value of (8). Let v^* be an optimal solution to (8) and define a vector $v' \in \mathbb{N}^n$ by $v_i' = \max\{0, -\lfloor v_i^* \rfloor\}$. Then the lattice program (6) with $z = u + v'$ is solved by $I(u,\tau)$. In fact, z is the optimal solution, and $v^* + v'$ is a feasible solution for the linear relaxation (7). The difference between the optimal solution values of the two programs is at least $c \cdot (u + v') - c \cdot (v^* + v') = c \cdot u - c \cdot v^*$. ∎

Proof of Theorem 2.1. In light of Lemma 2.4 and Proposition 2.3, we just need to show that if $I(u,\tau)$ and $I(u',\tau)$ are two standard ideals with $u \le u'$, then the optimal value of (8) is at most that of (8) with u replaced by u'. In order to do this we will reformulate these programs. Let B be an $n \times m$ matrix whose columns form a lattice basis of $\mathcal{L}$, and let $\{b_1,\ldots,b_n\} \subset \mathbb{Z}^m$ be the rows of B. The feasible solutions to (8) are in bijection with $\{t \in \mathbb{R}^m : b_i t \le u_i, i \in \tau\}$ via $t \mapsto v = u - Bt$. If we let $w := \sum_{i=1}^n c_i b_i$, then the following linear program is equivalent to (8) and has the same objective function value:

$$(10) \qquad \text{Maximize} \quad w \cdot t \quad \text{subject to} \quad b_i \cdot t \le u_i \quad \forall i \in \tau$$

The cost vector w is independent of u. If we replace u by u' in (10) then the feasible region increases but the objective function is unchanged. Hence the optimal value of (10) can only increase when replacing u by u'.   ■

Example 2.5. Theorem 2.1 suggests that we compute $\mathrm{gap}(\mathcal{L}, c)$ by solving finitely many linear programs (8), one for each irreducible component $I(u, \tau)$ of $M(\mathcal{L}, c)$. One difficulty is that the number of irreducible components can be exponential in the problem size. We illustrate this phenomenon with an example taken from [20, § 4]. For $r \geq 4$ let $\mathcal{L}_r$ be the lattice generated by

$$(r, r, r), \ (r-1, r+1, r-1) \text{ and } (0, 0, r-2) \text{ in } \mathbb{Z}^3.$$

The index of $\mathcal{L}_r$ in $\mathbb{Z}^3$ is $2r(r-2)$, so we are in the situation of Example 2.2. Let c be a cost vector that chooses the *degree lexicographically* smallest solution of (6). From the Gröbner basis in [20, Lemma 4.5] we find that

$$M(\mathcal{L}_r, c) = \bigcap_{\substack{1 \leq a \leq r-2 \\ b = 2r+1-a}} \langle x^a, y^b, z \rangle \ \cap \bigcap_{\substack{1 \leq a \leq r-3 \\ b = r-1-a}} \langle x, y^a, z^b \rangle.$$

The number of irreducible components is $2r - 5$. This is exponential in $O(\log(r))$, the bit complexity of the data. In the next section we demonstrate how the monomial ideals $M(\mathcal{L}, c)$ can be encoded more efficiently.   ■

3. Gap function as a short rational function

In this section we prove Theorem 1.2. The result extends easily to the lattice programs of Section 2, but for the sake of notational convenience, we present the relevant generating functions in the original setting of integer programs (1). Throughout the section we assume that d and n are fixed integers.

Let $\mathbb{N}A$ denote the semigroup in $\mathbb{Z}^d$ spanned by the columns $A_1, \ldots, A_n$ of the matrix A. The elements of $\mathbb{N}A$ are the feasible right-hand-side vectors of the integer programs we consider. Let $\mu_{IP} : \mathbb{N}A \mapsto \mathbb{Q}$ be the function whose value at b is the optimal value of the integer program (1), and let $\mu_{LP} : \mathbb{N}A \mapsto \mathbb{Q}$ be the corresponding optimal value function for the linear program (2). Since c is assumed to have rational coordinates, we can compute (in polynomial time) a *global denominator* $\Delta \in \mathbb{N}$ by multiplying the least common multiple of the maximal minors of A with the least common denominator of $c_1, c_2, \ldots, c_n$. This choice of Δ ensures that every value of the functions μ_{IP} or μ_{LP} is an integer multiple of Δ.

We are interested in the generating function

$$G(t_1, \ldots, t_d; s) := \sum_{b \in \mathbb{N}A} \mathbf{t}^b s^{\mu_{IP}(b) - \mu_{LP}(b)} \in \mathbb{Q}[s^{1/\Delta}][[\mathbb{N}A]].$$

The ambient ring is the completion of the semigroup algebra of $\mathbb{N}A$ with coefficients in the univariate polynomial ring $\mathbb{Q}[s^{1/\Delta}]$. We shall see that G is a rational series which can be represented by an element of

$$\mathbb{Q}(t_1,\ldots,t_d)[s^{1/\Delta}] \subset \mathbb{Q}(t_1,\ldots,t_d,s^{1/\Delta}).$$

Thus G is a polynomial in $s^{1/\Delta}$ whose coefficients are rational functions in $t_1,\ldots,t_d$. The degree of that univariate polynomial is the integer programming gap $\mathrm{gap}(A,c)$. We shall prove the following complexity result.

Theorem 3.1. *The rational function $G(t_1,\ldots,t_d;s)$ can be computed in polynomial time in the binary encoding of the matrix A and the cost vector c.*

What we are claiming is that $G(\mathbf{t};s)$ is a *short rational generating function* in the sense of Barvinok and Woods [3], and our proof is a direct application of their work. A different approach to the integer programming gap using generating functions and integer programming duality is presented by Lasserre in [15]. We illustrate the main point of Theorem 3.1 in an example.

Example 3.2. Let $d=1, n=2, a, b \in \mathbb{N}$ and consider the integer program

$$\text{minimize } u_1 \text{ subject to } u_1 + au_2 = b, \ u_1, u_2 \geq 0.$$

The optimal values are $\mu_{LP}(b)=0$ and $\mu_{IP}(b)=b \bmod a$. Hence

$$G(t,s) = \sum_{b \in \mathbb{N}} t^b s^{b \bmod a} = \sum_{i=0}^{a-1} \sum_{j=0}^{\infty} t^{i+ja} s^i = \sum_{i=0}^{a-1} (t^i/(1-t^a)) \cdot s^i.$$

This is a polynomial in s with a terms. Its expansion requires exponential space in the bit size $O(\log(a))$ of the given data $A=[1 \ a]$ and $c=[1 \ 0]$. On the other hand, clearly this gap polynomial is a short rational function

$$G(t,s) = \frac{1-s^a t^a}{(1-t^a)(1-st)}.$$

Our strategy is to compute $G(t,s)$ in polynomial time and then to extract $\mathrm{gap}(A,c)=\mathrm{degree}_s(G(t,s))=a-1$. ∎

One ingredient in the proof of Theorem 3.1 is the *Hadamard product* $*$ of two generating functions: if $g_1(\mathbf{x}) = \sum_u \beta_u \mathbf{x}^u$ and $g_2(\mathbf{x}) = \sum_v \gamma_v \mathbf{x}^v$ then $g_1(\mathbf{x}) * g_2(\mathbf{x})$ is the generating function $\sum \beta_u \gamma_u \mathbf{x}^u$. The proof of [3, Lemma 3.4] tells us that, if $g_1(\mathbf{x})$ and $g_2(\mathbf{x})$ are short rational functions, then their Hadamard product $g_1(\mathbf{x}) * g_2(\mathbf{x})$ can be computed in polynomial time. Another ingredient is the following lemma which is of independent interest.

Lemma 3.3. *The generating function for all the optimal points,*

$$H(\mathbf{x}) = \sum \{\mathbf{x}^u \ : \ u \text{ is optimal for (1) with } b = Au\}.$$

can be computed in polynomial time, in the binary encoding of A and c.

Proof. The proof is an adaptation of [3, § 7.3]. Without loss of generality we assume that $c \in \mathbb{Z}^n$. Let $S = \{u : x^u \in M(A,c)\}$ be the set of non-optimal points, and let $f(S;x_1,\dots,x_n) = \sum_{u\in S}\mathbf{x}^u$ be the generating function of S. This generating function is the $\mathbb{N}^n$-graded Hilbert series of the monomial ideal $M(A,c)$ and hence it is a rational function of the form $f(S;\mathbf{x})=g(S;\mathbf{x})/\prod_{i=1}^n(1-x_i)$, where $g(S;\mathbf{x})$ is a polynomial. In particular, we have

$$f(S;\mathbf{x}) \prod_{i=1}^{n}(1 - x_i) = g(S;\mathbf{x})$$

where $g(S;\mathbf{x})$ is a polynomial. In view of the identity

$$H(\mathbf{x}) = \frac{1 - g(S;\mathbf{x})}{\prod_{i=1}^n(1 - x_i)},$$

it suffices to show that $g(S;\mathbf{x})$ can be computed in polynomial time.

We let $L := (n - d)D(A)$ where $D(A)$ is the maximum of the absolute values of the maximal minors of A. Since we fix d and n, the bit size of L is a polynomial in the bit size of the data. Theorem 4.7 of [18] implies that if $\mathbf{x}^m$ is a term in $g(S;\mathbf{x})$ then $m_i \le L$ for $i = 1,\dots,n$. We let $\Lambda := \{(u_1,\dots,u_n)\in \mathbb{N}^n : u_i \le L, \forall i\}$ and $f(\Lambda;\mathbf{x})$ its generating function. Furthermore let

$$C := \big\{(u,v) \in \mathbb{N}^{2n} \ : \ Au = Av, \ cu \ge cv + 1, \ \text{and} \ u_i \le L, \ v_i \le 2L, \ \forall i\big\}$$

The projection of C onto the first n coordinates will be denoted by S', and we claim that $S' = S \cap \Lambda$. The inclusion $S' \subseteq S \cap \Lambda$ is clear. For the other inclusion let $u \in S \cap \Lambda$. The theory of Gröbner bases of toric ideals [18] implies that there exists a Gröbner basis element that improves the non-optimal solution u. In other words, there exists $(u',v') \in \mathbb{N}^{2n}$ with $u' \le u$, $Au' = Av'$ and $cu' \ge cv' + 1$, and where $u_i'v_i' = 0$, $u_i' \le L$ and $v_i' \le L$ for $i = 1,\dots,n$. Now we let $v = u - u' + v'$ and observe that $(u,v)\in C$. Since S' is the projection of all lattice points in a polytope, Theorem 1.7 in [3] implies that $f(S';\mathbf{x})=\sum_{u\in S'}\mathbf{x}^u$, and hence $g(S';\mathbf{x}) = f(S';\mathbf{x})\prod_{i=1}^n(1-x_i)$, can be computed in polynomial time. The claim we proved above says that

$$f(S;\mathbf{x}) = f(S';\mathbf{x}) + \sum_{u\in S\backslash\Lambda} \mathbf{x}^u,$$

and this implies $g(S; \mathbf{x}) = g(S'; \mathbf{x}) + h(\mathbf{x})$, where $h(\mathbf{x})$ is a series none of whose terms has its exponent vector in Λ. Now we conclude that

$$g(S; \mathbf{x}) = g(S; \mathbf{x}) * f(\Lambda; \mathbf{x}) = g(S'; \mathbf{x}) * f(\Lambda; \mathbf{x}).$$

The Hadamard product on the right can be computed in polynomial time. ∎

Proof of Theorem 3.1. We replace each variable x_i by $\mathbf{t}^{A_i} s^{c_i}$ in the rational function $H(\mathbf{x})$. This monomial substitution can be done in polynomial time [3, Theorem 2.6]. The result is the short rational generating function

$$H_{IP}(t_1, \dots, t_d; s) = \sum_{b \in \mathbb{N}A} \mathbf{t}^b s^{\mu_{IP}(b)}.$$

The last series that is left to compute is

$$H_{LP}(t_1, \dots, t_d; s) = \sum_{b \in \mathbb{N}A} \mathbf{t}^b s^{-\mu_{LP}(b)},$$

since $G(\mathbf{t}; s) = H_{IP}(\mathbf{t}; s) *_{\mathbf{t}} H_{LP}(\mathbf{t}; s)$. Let $\sigma = \{i_1, \dots, i_d\} \subset \{1, \dots, n\}$ be an optimal basis of (2) for some b in $\mathbb{N}A$. The number of optimal bases σ is constant since n and d are fixed. The union of $S_\sigma := \mathbb{N}A \cap \mathbb{R}_{\geq 0}\{A_i : i \in \sigma\}$ as σ varies over all optimal bases is equal to $\mathbb{N}A$. The generating function $f(S_\sigma; \mathbf{t}) = \sum_{b \in S_\sigma} \mathbf{t}^b$ can be computed in polynomial time since S_σ is the set of lattice points in a rational polyhedral cone [2]. We let $\hat{c} \in \mathbb{R}^d$ be the unique vector such that $(A^t \cdot \hat{c})_i = c_i$ for $i \in \sigma$. Now the generating function $g_\sigma(\mathbf{t}; s) = \sum_{b \in S_\sigma} \mathbf{t}^b s^{-\mu_{LP}(b)}$ is obtained from $f(S_\sigma; \mathbf{t})$ by the monomial substitution $t_i = t_i s^{-\hat{c}_i}$ for $i = 1, \dots, d$. Finally, we use [3, Corollary 3.7] to compute $H_{LP}(\mathbf{t}; s)$ by patching the series $g_\sigma(\mathbf{t}; s)$ for the various bases σ. ∎

We now show that $\mathrm{gap}(A, c)$, which is the degree of $G(\mathbf{t}; s)$ as a polynomial in s, can be extracted in polynomial time. This uses the following lemma.

Lemma 3.4. *Let $f(\mathbf{t}; s) \in \mathbb{Q}(t_1, \dots, t_d)[s]$ be a short non-zero rational function and K a known upper bound on $\deg_s(f(\mathbf{t}; s))$. Then this degree can be computed with $\log(K)$ Hadamard products of $f(\mathbf{t}; s)$ with polynomials in s.*

Proof. Without loss of generality we assume $K = 2^k$ for some $k \in \mathbb{N}$. Let

$$g_{[p,r]}(s) = \sum_{i=p}^{r} s^i = \frac{s^p - s^{r+1}}{1 - s},$$

and use the following binary search algorithm:

`findDegree`$(f(\mathbf{t};s),p,r)$

0. If $p=r$ return p.
1. If $f(\mathbf{t};s) * g_{[p,r]}(s)$ is not identically zero then
   `findDegree`$(f(\mathbf{t};s),\lfloor (r+p)/2 \rfloor,r)$.
2. If $f(\mathbf{t};s) * g_{[p,r]}(s)$ is identically zero then
   `findDegree`$(f(\mathbf{t};s),\lfloor p/2 \rfloor,p-1)$.

The call `findDegree`$(f(\mathbf{t};s),0,K)$ takes at most $\log(K)$ steps to find the degree. Zero testing can be done in polynomial time, though it needs great care to achieve this results. We refer to Section 2.2 of [6] for details of accomplishing this. Note that this Hadamard product is a polynomial in s whose coefficients have expansions of the form $\sum_{b \in \Gamma} \mathbf{t}^b$ for some set Γ. This means that the substitution of the above positive vector gives a positive number unless the Hadamard product is identically zero. ∎

Proof of Theorem 1.2. By Theorem 3.1 we can compute the rational function $G(\mathbf{t};s)$ in polynomial time. The degree of $G(\mathbf{t};s)$ viewed as a polynomial in s is $\mathrm{gap}(A,c)$. Theorem 17.2 of [17] implies that $\mathrm{gap}(A,c) \leq nD(A)\sum_{i=1}^{n}|c_i|$. Let K be this upper bound. We observe that $\log(K)$ is a polynomial in the bit size of the data. We can hence use Lemma 3.4 to find $\mathrm{gap}(A,c)$. ∎

4. An Application to Algebraic Statistics

We present an application to the statistical theory of *disclosure limitation*. See [5] and [8] and the references therein. Suppose we are given data in the form of an n-dimensional table of nonnegative integers. The aim is to release some marginals of the table but not the table's entries themselves. If the range of possible values that a particular entry can attain in any table satisfying the released marginals is too narrow, or even worse, consists of the unique value of that entry in the actual table, then this entry may be exposed. This shows the importance of determining tight integer upper and lower bounds for each entry in a given table. A choice of marginals corresponds to a simplicial complex on $\{1,2,\dots,n\}$ and can be represented by a zero-one matrix A, as described in [10, § 1]. In statistical language, the matrix A specifies a *hierarchical model* for a *contingency table* with n factors.

The *table entry security problem* can be formally stated as follows: suppose u is a table with nonnegative integer entries, where the marginals are computed according to a fixed hierarchical model A and let $u_{i_1 i_2 \cdots i_n}$ be a particular cell for such tables. Compute optimal lower and upper bounds L and U such that $L \leq u_{i_1 i_2 \cdots i_n} \leq U$ for all tables with the same marginals as u.

This problem is an integer program (1): minimize (or maximize) $u_{i_1 i_2 \cdots i_n}$ over all tables with nonnegative *integer* entries subject to fixing the marginals. In view of the difficulty of solving integer programs exactly, various researchers resorted to solving the linear programming relaxation (2) instead: minimize (or maximize) $u_{i_1 i_2 \cdots i_n}$ over all tables with nonnegative *real* entries subject to fixing the marginals. This relaxation is tractable, but it usually fails to deliver the exact integers L and U. One faces the problem of estimating the integer programming gap for the table security problem.

We give the precise definition of the relevant matrices A. Consider $d_1 \times \cdots \times d_n$-tables with entries $u_{i_1 i_2 \cdots i_n}$ where $1 \leq i_j \leq d_j$. We fix a hierarchical model by specifying a collection of subsets $F_1, \ldots, F_k$ of $\{1, \ldots, n\}$. The marginals of our table are computed with respect to these subsets. If $F_i = \{j_1, \ldots, j_s\}$ then the F_i-marginal is a $d_{j_1} \times \cdots \times d_{j_s}$ table b with entries

$$(11) \qquad b_{k_1 \cdots k_s} = \sum_{i_{j_1} = k_1, \ldots, i_{j_s} = k_s} u_{i_1 \cdots i_n}.$$

Example 4.1. The classical *transportation problem* corresponds to $d_1 \times d_2$-tables where the marginals are computed with respect to $F_1 = \{1\}$ and $F_2 = \{2\}$. The *three-dimensional* transportation problem concerns $d_1 \times d_2 \times d_3$-tables with $F_1 = \{1, 2\}$, $F_2 = \{1, 3\}$, and $F_3 = \{2, 3\}$. The marginals are

$$b_{ij} = \sum_k u_{ijk}, \quad b_{ik} = \sum_j u_{ijk}, \quad b_{jk} = \sum_i u_{ijk}.$$

For a discussion from the Gröbner basis perspective see [18, § 14.C]. ∎

We define A to be the zero-one matrix with $d_1 d_2 \cdots d_n$ columns that corresponds to the linear map that computes the marginals of tables. We let u be the vector of variables representing the cell entries. Then $A \cdot u$ is the vector of the k lower-dimensional tables computed as in (11). There is a transitive symmetry group acting on the columns of A, so it suffices to examine the particular cell entry $u_{11 \cdots 1}$ which corresponds to the first column of A. The table entry security problem is the pair of integer programs

(12) Minimize (Maximize) $u_{11 \cdots 1}$ subject to $A \cdot u = b, \ u \geq 0, \ u$ integral.

Our Theorem 1.1 gives an exact formula for the integer programming gap of these problems, which we abbreviate by $\mathrm{gap}_-(A)$ and $\mathrm{gap}_+(A)$ respectively. Thus $\mathrm{gap}_+(A)$ is the worst error one gets when using linear programming in computing the bound U for any $d_1 \times \cdots \times d_n$-table with any fixed margins b.

We illustrate our results for $2 \times 2 \times 2 \times 2$-tables (u_{ijkl}). The K_4-*model* is specified by taking all six two-dimensional margins $F_1 = \{1,2\}$, $F_2 = \{1,3\}$, $F_3 = \{1,4\}$, $F_4 = \{2,3\}$, $F_5 = \{2,4\}$, $F_6 = \{3,4\}$. The zero-one matrix A for the K_4-model has 24 rows and 16 columns:

$$A = \begin{bmatrix}
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1
\end{bmatrix}$$

Here the cell entries are ordered lexicographically, $u_{1111}, u_{1112}, \ldots, u_{2222}$.

We have the following computational result.

Proposition 4.2. *Every hierarchical model for $2 \times 2 \times 2 \times 2$-tables satisfies*

$$\mathrm{gap}_+(A) \le 5/3 = 1.666666\ldots,$$

and this bound is attained for the K_4-model, that is, for the matrix A above.

Proof. We show that $\mathrm{gap}_+(A) = 5/3$ for the K_4-model. Similar computations show that $\mathrm{gap}_+(A) < 5/3$ for all other simplicial complexes on $\{1,2,3,4\}$.

The monomial ideal $M(A,c)$ where A is the above matrix and $c = [-1,0,\ldots,0]$ has 61 minimal generators. Two of them are

$$x_{1112}^3 x_{1221} x_{1222} x_{2121} x_{2122} x_{2211} x_{2212} \quad \text{and} \quad x_{1122} x_{1212} x_{1221} x_{2111} x_{2222}^2.$$

This ideal has 139 irreducible components. One of these components is

$$I(u,\tau) = \big\langle x_{1112}^2, x_{1121}^2, x_{1122}, x_{1211}^2, x_{1212}, x_{1221}, x_{1222},$$
$$x_{2111}^2, x_{2112}, x_{2121}, x_{2122}, x_{2211}, x_{2212}, x_{2221}, x_{2222}^2 \big\rangle.$$

Here the non-zero components of u are $u_{1112}=u_{1121}=u_{1211}=u_{2111}=u_{2222}=1$. The gap value of $I(u,\tau)$ is $5/3$. We also give the optimal solution v for the linear program (2) with $b=A\cdot u$:

$$v_{1112}=v_{1121}=v_{1211}=v_{2111}=v_{2222}=0,$$
$$v_{1122}=v_{1212}=v_{1221}=v_{1222}=v_{2112}=1/3,$$
$$v_{2121}=v_{2122}=v_{2211}=v_{2212}=v_{2221}=1/3, \qquad v_{1111}=5/3. \qquad \blacksquare$$

Remark. The above example can be generalized to show that the integer programming gap can be large. In particular, there exist table entry security problems where the LP relaxation would indicats that a particular cell entry is secure whereas in reality it is not. For this we use n-dimensional $2\times\cdots\times 2$ tables where all two-dimensional marginals are computed (the binary K_n model). Proposition 2.7 in [7] implies that the table with

$$u_{11\ldots 12}=u_{11\ldots 21}=\cdots=u_{21\ldots 11}=1 \text{ and } u_{22\ldots 22}=n-3$$

and all the other entries equal to zero is the only table with the integer entries that has the corresponding marginal vector $A\cdot u=b$. This means that the integer program's optimal value is equal to zero. However, here is a fractional table with the same marginals:

$$v_{11\ldots 12}=v_{11\ldots 21}=\cdots=v_{21\ldots 11}=0, \text{ and } v_{22\ldots 22}=n-4$$

with $v_{11\ldots 11}=(n-3)+(n-2)(2^{n-2}-1)^{-1}$ and all the other entries equal to $(2^{n-2}-1)^{-1}$. Hence the LP gives the lower bound $(n-3)+(n-2)(2^{n-2}-1)^{-1}$. This number is a lower bound for the gap. It tends to infinity for large n.

The following corollary was pointed out to us by Rekha Thomas.

Corollary 4.3. *Let $M(A,c)$ be the monomial ideal corresponding to the minimization problem (12). Then $\mathrm{gap}_-(A,c)+1$ is at least the maximum degree of $x_{11\ldots 1}$ in any minimal generator of $M(A,c)$.*

Proof. This statement is equivalent to

$$\mathrm{gap}_-(A) \geq \max\{u_{11\ldots 1} : I(u,\tau) \text{ irreducible component of } M(A,c)\}.$$

Let $I(u,\tau)$ be an irreducible component such that $u_{11\ldots 1} \geq 1$. Since $c=[1,0,\ldots,0]$, the objective function of the program (10) corresponding to this component would be $b_{11\ldots 1}t$. Moreover, by Theorem 2.5 in [12], the inequality $b_{11\ldots 1}\cdot t \leq u_{11\ldots 1}$ is a facet of the feasible region of (10). Hence the optimal solution value is $u_{11\ldots 1}$. Now Theorem 1.1 gives the result. $\blacksquare$

Remark. In the statistics literature there are various approaches to estimate L and U of the table security problem. For general hierarchical models an iterative algorithm for such an estimation is given in [1]. A detailed analysis for *decomposable models* is given by Dobra and Fienberg [8].

5. The gap fan

In this section we allow the cost function c to vary in the programs (1) or (6). For each fixed matrix A (resp. fixed lattice $\mathcal{L}$) we thus get a function

$$(13) \qquad \operatorname{gap}_A : \mathbb{R}^n \longrightarrow \mathbb{R}, \ c \mapsto \operatorname{gap}(A, c).$$

Our goal is to show that this function is piecewise linear, and the natural pieces on which the function is linear form a fan which we call the *gap fan*.

Consider the *Gröbner fan* of the matrix A. Following [18] and [19], this is the coarsest polyhedral fan in $\mathbb{R}^n$ on which the map $c \mapsto M(A, c)$ is constant. Efficient software packages for computing the Gröbner fan are described in [13] and [14]. The gap fan will be a refinement of the Gröbner fan. Hence it suffices to describe the gap fan on each Gröbner cone $\mathcal{K}$ separately.

We fix a maximal cone $\mathcal{K}$ of the Gröbner fan. By the results of [19], the polyhedral cone $\mathcal{K}$ consists of cost vectors such that the optimal solutions of all integer programs (1) are constant as the right-hand-side vector b varies. There exists a monomial ideal M in $\mathbb{R}[x_1, \ldots, x_n]$ such that

$$\mathcal{K} = \{c \in \mathbb{R}^n \ : \ M(A, c) = M\}.$$

Let $\{I(u, \tau)\}$ be the finite set of all irreducible components of the monomial ideal M. For each such component let $v_{u,\tau}$ be the optimal solution to the linear program (3). It is not hard to see that these optimal solutions are also constant as c varies in $\mathcal{K}$. Now we define the following polyhedral cone:

$$(14) \qquad \mathcal{G} := \{(c, t) \in \mathbb{R}^{n+1} \ : \ c \in \mathcal{K} \text{ and } c \cdot (u - v_{u,\tau}) \leq t \ \forall I(u, \tau)\}$$

Theorem 5.1. *The projection of the lower envelope of $\mathcal{G}$ onto $\mathcal{K}$ is a polyhedral subdivision of $\mathcal{K}$. The gap function is linear on each face of this subdivision.*

Proof. The facets of the lower envelope of $\mathcal{G}$ are of the form $\{(c, t) \in \mathbb{R}^{n+1} : c \cdot (u - v_{u,\tau}) = t\}$ where t is the optimal value of the program (3) for all c such that (c, t) is on this facet. For such pairs (c, t) we have $t = \operatorname{gap}(A, c)$, by Theorem 1.1. Hence the lower envelope of $\mathcal{G}$ is the graph of the gap function (13) over $\mathcal{K}$. Now, clearly, the projection of the lower envelope onto $\mathcal{K}$ is a polyhedral subdivision of $\mathcal{K}$, and by its construction, the gap function (13) is linear on each cone in this subdivision of $\mathcal{K}$. ∎

We define *gap fan* as the refinement of the Gröbner fan of A, where each Gröbner cone $\mathcal{K}$ is subdivided as in Theorem 5.1. Our discussion implies:

Corollary 5.2. *The gap function (13) is a piecewise linear function on $\mathbb{R}^n$. It is linear on the cones of the gap fan, but generally not on the Gröbner fan.*

Example 5.3. Let us return to the example in the Introduction but now with varying cost function. Our problem is to make change using pennies, nickels, dimes and quarters, where the number of coins is fixed. The Gröbner fan of the corresponding 2×4-matrix A has seven maximal cones. These cones and the corresponding ideals can be computed using `TiGERS` [13] or `CaTS` [14].

The gap fan has eight cones. Exactly one cone of the Gröbner fan is divided into two cones. It is the one corresponding to the ideal in (5). This Gröbner cone is defined by the inequalities

$$3n + q \geq 4d \quad \text{and} \quad 8d \leq 5p + 3q.$$

The hyperplane defined by

$$305p - 135n - 308d + 138q = 0$$

splits this cone into two pieces. On the positive side of the hyperplane the gap(A, c) is given by the irreducible component $\langle p^5, n^3 \rangle$, and on the negative side it is given by $\langle n^6, d^4, q \rangle$. We list the irreducible components of all seven initial ideals and the winning irreducible component for each of them:

$M(A, c)$	Winning component(s)
$\langle p^5, d^4 \rangle$	$\langle p^5, d^4 \rangle$
$\langle p^5, q \rangle \cap \langle p^5, n^3 \rangle \cap \langle d^4, q \rangle$	$\langle p^5, n^3 \rangle$
$\langle p^5, n^3 \rangle \cap \langle n^9, q \rangle$	$\langle p^5, n^3 \rangle$
$\langle p^5, n^3 \rangle \cap \langle n^6, q \rangle \cap \langle n^3, q^2 \rangle$	$\langle p^5, n^3 \rangle$
$\langle p^5, n^3 \rangle \cap \langle n^6, d^4, q \rangle \cap \langle n^3, q^3 \rangle$	$\langle p^5, n^3 \rangle$ $\langle n^6, d^4, q \rangle$
$\langle n^6, d^4, q \rangle \cap \langle n^3, d^8 \rangle$	$\langle n^6, d^4, q \rangle$
$\langle n^6, d^4 \rangle$	$\langle n^6, d^4 \rangle$

References

[1] L. BUZZIGOLI and A. GUSTI: An algorithm to calculate the lower and upper bounds of the elements of an array given its marginals, in *Statistical Data Protection Proceedings*, Eurostat, Luxembourg (1999), pp. 131–147.

[2] A. BARVINOK and J. E. POMMERSHEIM: An algorithmic theory of lattice points in polyhedra, in *New Perspectives in Algebraic Combinatorics*, MSRI Publications **38** (1999), pp. 91–147.

[3] A. BARVINOK and K. WOODS: Short rational generating functions for lattice point problems, *Journal of the American Math. Soc.* **16** (2003), 957–979.

[4] D. COX, J. LITTLE and D. O'SHEA: *Using Algebraic Geometry*, Graduate Texts in Mathematics **185**, Springer-Verlag, New York, 1998.

[5] L. Cox and J. George: Controlled rounding for tables with subtotals, *Annals of Operations Research* **20** (1989), 141–157.

[6] J. A. De Loera, R. Hemmecke, J. Tauzer and R. Yoshida: Effective lattice point counting in rational convex polytopes, *Symbolic Computation* **38(4)** (2004), 1273–1302.

[7] M. Develin and S. Sullivant: Markov bases of binary graph models, *Annals of Combinatorics* **7** (2003), 441–466; math.CO/0308280.

[8] A. Dobra and S. Fienberg: Bounds for cell entries in contingency tables given marginal totals and decomposable graphs, *Proc. Natl. Acad. Sci. USA* **97** (2000), 11885–11892.

[9] D. Eisenbud, D. Grayson, M. Stillman and B. Sturmfels: *Mathematical Computations with* `Macaulay2`, Algorithms and Computation in Mathematics, Vol. **8**, Springer-Verlag, Heidelberg, 2001.

[10] S. Hoşten and S. Sullivant: Gröbner bases and polyhedral geometry of reducible and cyclic models, *J. Combinatorial Theory, Ser. A* **100** (2002), 277–301.

[11] S. Hoşten and R. Thomas: Standard pairs and group relaxations in integer programming, *J. Pure Appl. Algebra* **139** (1999), 133–157.

[12] S. Hoşten and R. Thomas: The associated primes of initial ideals of lattice ideals, *Math. Res. Lett.* **6** (1999), 83–97.

[13] B. Huber and R. Thomas: Computing Gröbner fans of toric ideals, *Experimental Mathematics* **9** (2000), 321–331.

[14] A. Jensen: `CaTS` – A computer program for computing the Gröbner fan of a toric ideal, http://www.soopadoopa.dk/anders/cats/cats.html.

[15] J. B. Lasserre: Duality and a Farkas lemma for integer programs, in *Optimization: Structure and Applications* (E. Hunt and C. E. M. Pearce, editors), Applied Optimization Series, Kluwer Academic Publishers, 2003.

[16] M. Saito, B. Sturmfels and N. Takayama: *Gröbner Deformations of Hypergeometric Differential Equations*, Algorithms and Computation in Mathematics **6**, Springer-Verlag, Berlin, 2000.

[17] A. Schrijver: *Theory of Linear and Integer Programming*, Wiley Interscience, Chichester, 1986.

[18] B. Sturmfels: *Gröbner Bases and Convex Polytopes*, University Lecture Series **8**, American Mathematical Society, 1995.

[19] B. Sturmfels and R. Thomas: Variation of cost functions in integer programming, *Mathematical Programming* **77** (1997), 357–387.

[20] B. Sturmfels, R. Weismantel and G. Ziegler: Gröbner bases of lattices, corner polyhedra, and integer programming; *Beiträge zur Algebra und Geometrie* **36** (1995), 281–298.

[21] R. R. Thomas: The structure of group relaxations, to appear in *Handbook of Discrete Optimization* (eds.: K. Aardal, G. Nemhauser, R. Weismantel), 2003, http://www.math.washington.edu/∼thomas/articles.html.

Serkan Hoşten

Department of Mathematics
San Francisco State University
1600 Holloway Avenue
San Francisco, CA 94530, USA
serkan@math.sfsu.edu

Bernd Sturmfels

Department of Mathematics
University of California
Berkeley, CA 94720, USA
bernd@math.berkeley.edu