AIAA JOURNAL
Vol. 36, No. 10, October 1998

New Adaptive Move-Limit Management Strategy for Approximate Optimization, Part 2

B. A. Wujek*
Engineous Software, Inc., Morrisville, North Carolina 27560
and
J. E. Renaud†
University of Notre Dame, Notre Dame, Indiana 46556-5637

Approximations play an important role in multidisciplinary design optimization by offering system behavior information at a relatively low cost. Most approximate optimization strategies are sequential, in which an optimization of an approximate problem subject to design variable move limits is iteratively repeated until convergence. The move limits are imposed to restrict the optimization to regions of the design space in which the approximations provide meaningful information. To ensure convergence of the sequence of approximate optimizations to a Karush–Kuhn–Tucker solution a move-limit management strategy is required. In a companion paper, issues of move-limit management are reviewed and a new adaptive strategy for move-limit management is developed (Wujek, B. A., and Renaud, J. E., "New Adaptive Move-Limit Management Strategy for Approximate Optimization, Part 1," *AIAA Journal*, Vol. 36, No. 10, 1998, pp. 1911–1921). With its basis in the provably convergent trust region methodology, the trust region ratio approximation method (TRAM) strategy utilizes available gradient information and employs a backtracking process using various two-point approximation techniques to provide a flexible move-limit adjustment factor. The TRAM strategy is successfully implemented in application to several multidisciplinary design optimization test problems. In addition, a comprehensive study comparing the performance of the TRAM strategy to existing move-limit strategies is conducted. Results indicate that application of the TRAM strategy results in increased efficiency for approximate optimization processes. These implementation studies highlight the ability of the TRAM strategy to control the amount of approximation error and efficiently manage the convergence to a Karush–Kuhn–Tucker solution.

I. Introduction

THE use of approximations to represent the design space is essential to the efficiency of multidisciplinary design optimization (MDO) algorithms. Approximations provide information about the system necessary for the optimization process without the cost of executing CPU-intensive analysis tools. Moreover, the use of approximations allows for the temporary decoupling of disciplines, which avoids the constant transfer of information among disciplines required during an iterative system analysis. Through the use of design space approximations, optimization of large, complex systems is made more practicable. It is important that these approximations accurately portray the design space so that the infeasible region is avoided and the design objective is continuously improving. These approximations will tend to stray from the actual system response surface as the design moves away from the data point(s) about which the approximation was formed. Therefore, design variable move limits are imposed to restrict the approximate optimization to regions of the design space in which the approximations are accurate. After each sequence of approximate optimization, the approximations of system behavior are updated with new information about the current design. Thus, many iterations of such algorithms may be required before convergence of the optimization process is achieved, and every additional iteration adds to the cost of the process. In light of this, a primary concern in developing an approximate optimization strategy is the proper choice of a move-limit management strategy.

The theoretical basis and implementation details of a new move-limit management strategy are developed in a companion paper.[1] This new move-limit management method is based on approximating the trust region ratio ρ and is referred to as the trust region ratio approximation method (TRAM) strategy. With its basis in the provably convergent trust region methodology, the TRAM strategy utilizes available gradient information and employs a backtracking process using various two-point approximation techniques to provide a flexible move-limit adjustment factor. The TRAM strategy provides a more flexible move-limit adjustment factor than direct trust region methods through the use of trust region ratio sensitivities. Coupled with a new approximation-based point rejection scheme, the use of the TRAM strategy results in less approximation error and leads to faster algorithm convergence. The development of the TRAM strategy provides improved efficiency in the execution of approximate optimization algorithms.

The flowchart in Fig. 1 details the TRAM strategy. The quantities that must be obtained prior to executing the TRAM strategy are listed preceding the flowchart. These quantities require that a system analysis and a sensitivity analysis be carried out on the current design iterate and that similar information be available for the previous design iterate. These procedures are common to approximate optimization algorithms so that no extra analysis burden is placed on the designer. The only additional cost associated with implementing the TRAM strategy is in executing the root solving algorithms to obtain both r (in step 3) and α_{acc} in step 4 (see Ref. 1 for details of the algorithm steps). However, because these are one-dimensional problems of simple analytic functions with available analytic gradients, the cost can be considered rather insignificant relative to the cost of other procedures.

The TRAM strategy is generally applicable to most approximate optimization algorithms. In this paper it is implemented and studied in application to the concurrent subspace optimization (CSSO) algorithm.[2] It is incorporated in a move-limit submodule located within the algorithm flowchart, as shown in Fig. 2. The move-limit submodule is a general standalone component of the algorithm, which serves to automatically manage the move limits for the approximate optimization module based on whatever strategy is desired. Thus, other strategies besides the TRAM strategy can easily be

Received Feb. 6, 1998; presented as Paper 98-1966 at the AIAA/ASME/ASCE/AHS/ASC 39th Structures, Structural Dynamics, and Materials Conference, Long Beach, CA, April 20–23, 1998; revision received June 1, 1998; accepted for publication June 2, 1998.

*Advanced Technology Developer. Member AIAA.
†Assistant Professor, Department of Aerospace and Mechanical Engineering. Member AIAA.

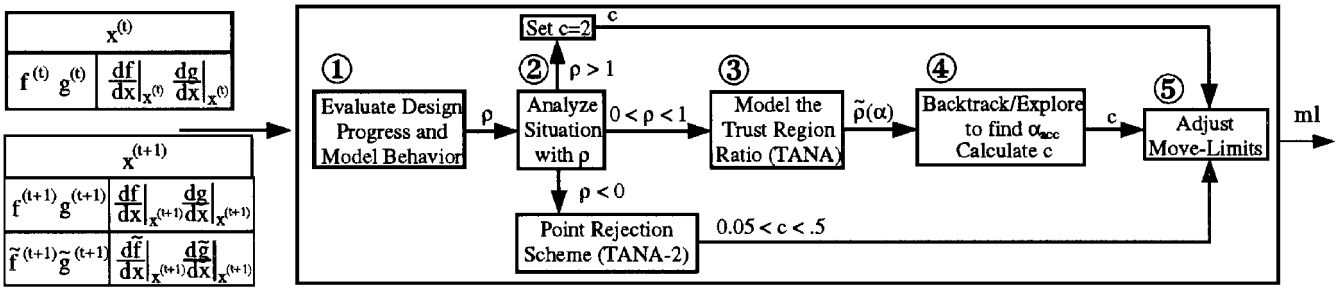

Fig. 1 TRAM implementation flowchart.

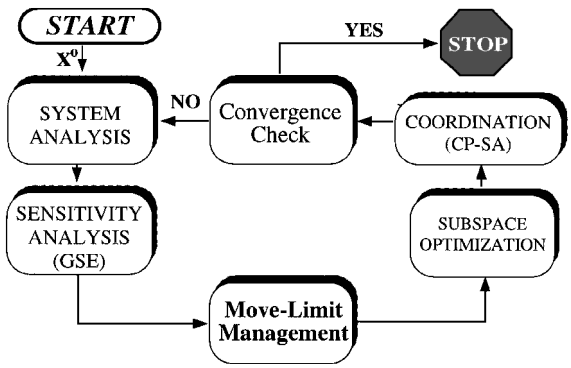

Fig. 2 Move-limit management submodule for CSSO.

included as options for the designer (the MDO framework NDOPT[3] is used in this research). This feature is exploited in the implementation studies discussed in the following section.

II. Implementation Studies

As a means of evaluating the efficiency and effectiveness of the TRAM move-limit strategy, the procedure presented in the Introduction was incorporated in a move-limit algorithm submodule as discussed and implemented on four test problems within the framework of CSSO and other approximate optimization algorithms. For comparison purposes, a number of the methods reviewed earlier[1] were also coded and tested. However, some of the methods were not suitable for a general implementation in CSSO due to the specific nature of their methodologies. For instance, the strategies of Fadel et al.[4] and Grignon and Fadel[5] are based on the use of a restricted two-point exponential approximation for the coordination procedure; their use would require restructuring of the system coordination procedure in CSSO. On the other hand, the prescribed reduction scheme of Pourazady and Fu[6] (P–F) and the strategies of Thomas et al.,[7] Bloebaum et al.,[8] and Chen[9] can all be easily incorporated in the move-limit submodule because they only require information that is readily available (f, $\boldsymbol{g}$, and/or their gradients). Also, because the direct trust region (DTR) approach is the basis for the TRAM strategy, it too is included for testing. It must be noted that in implementing these strategies, no attempt was made to fine tune the parameters associated with the specific methods. Indeed, it is this fine tuning that should be carried out automatically by a move-limit strategy. Sensitivity to specific parameters and inability to make necessary adjustments automatically are seen as limitations of a move-limit strategy.

To present the results of all of the strategies in a concise organized fashion, the following categories of strategies are defined: situational, comprising TRAM, DTR, Bloebaum et al.,[8] and Thomas et al.[7]; constant, comprising constant A%, constant B%, P–F A%, and P–F B%; and Chen,[9] comprising Chen's methods I–IV. The A% and B% in the constant strategies refer to different levels of initial move-limit values that are specifically noted in each problem. These different initial move limits were required because the effectiveness of the constant strategies are highly dependent on them. For the situational strategies, initial move limits were set at reasonable percentages ($\approx$10–30%) that are specified for each problem.

Because the move-limit management study is implemented specifically within the CSSO algorithm, another issue that must be considered is the setting of move limits for subspace optimization procedures. Although the importance of proper subspace move-limit management cannot be overlooked, the present study is focused on system-level move limits. Proper subspace move-limit management is left for future study. It is reasonable to set the subspace move limits to be equal to those prescribed for the system level so that data for approximating the Hessian is obtained throughout this constrained region. This is the procedure employed in this study.

In selecting the test problems, certain characteristics were sought. It was desired to first test the strategies on problems of low dimensionality (few design variables and constraints) so that the design space could be easily illustrated and studied and the results more easily interpreted. With the intricacies of the strategies observed on the smaller problems, problems of higher dimensionality can then be used to test the effectiveness in larger design spaces (more design variables) with more constraints. In any size problem, a desired feature is complexity and nonlinearity in the system states, objective, and constraints so that significant error is introduced by the use of approximations. In the remainder of this paper, the problems used to test the move-limit strategies are briefly described and the corresponding results are discussed.

III. Test Problems and Results

Barnes's Problem

This problem, originally formulated as part of a M.S. Thesis by Barnes,[10] is a purely mathematical two-dimensional problem with a highly complicated nonlinear objective function and three nonlinear constraints. To formulate the problem as an MDO problem, five states were defined to represent various components of the objective function and constraints, and two contributing analyses (CAs) were created to calculate these states. The system level optimization problem in standard form is as follows:

Minimize:

$$f(\boldsymbol{x},\boldsymbol{y}) = a_1 + a_2x_1 + a_3y_4 + a_4y_4x_1 + a_5y_4^2 + a_6x_2 + a_7y_1 + a_8x_1y_1 + a_9y_1y_4 + a_{10}y_2y_4 + a_{11}y_3 + a_{12}x_2y_3 + a_{13}y_3^2 + [a_{14}/(x_2+1)] + a_{15}y_3y_4 + a_{16}y_1y_4x_2 + a_{17}y_1y_3y_4 + a_{18}x_1y_3 + a_{19}y_1y_3 + a_{20}\,e^{a_{21}y_1}$$

Subject to:

$$g_1 = (y_1/700) - 1 \geq 0, \qquad g_2 = (x_2/5) - \left(y_4/25^2\right) \geq 0$$

$$g_3 = (y_5 - 1)^2 - [(x_1/500) - 0.11] \geq 0$$

$$0.0 \leq x_1 \leq 75.0, \qquad 0.0 \leq x_2 \leq 65.0$$

where the coefficients a_i are listed in Ref. 10 and the states are calculated by CA_1 as

$$y_1 = x_1x_2, \qquad y_3 = x_2^2$$

and by CA_2 as

$$y_2 = y_1x_1, \qquad y_4 = x_1^2, \qquad y_5 = x_2/50$$

Although the CAs are structured such that they are characterized by only feedforward coupling, the problem is formulated so that each

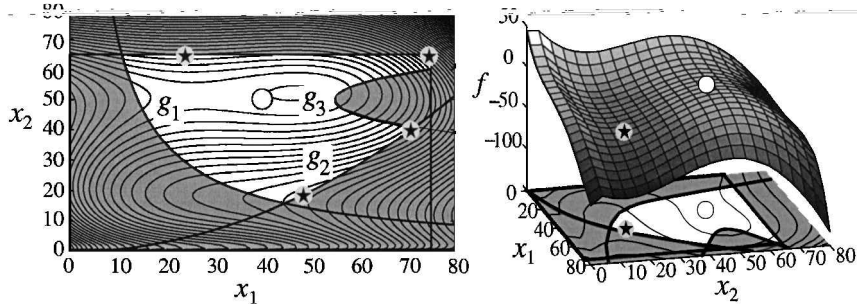

Fig. 3 Design space for Barnes's[10] problem.

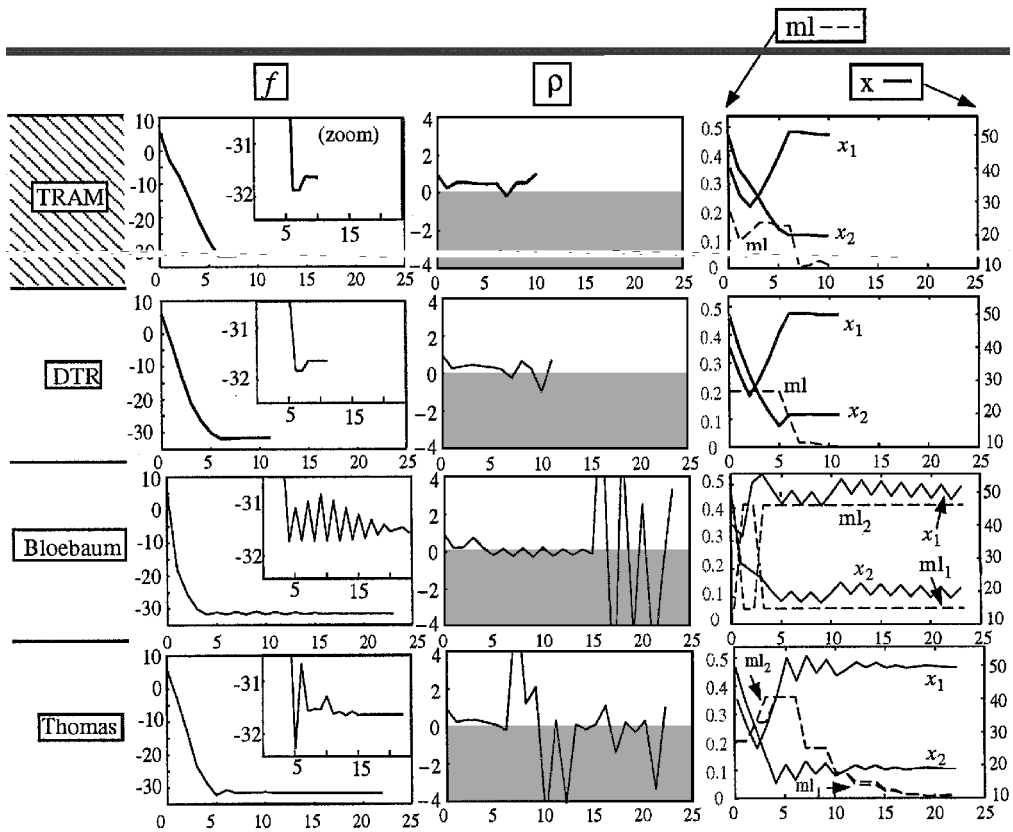

Fig. 4 Convergence plots for Barnes's[10] problem: situational strategies.

CA contributes some portion of the objective and constraints. A consequence of this formulation is that obtaining good approximations can be difficult, especially for the objective function.

Whereas the interactions among the design variables in the objective is quite complex, only two design variables are defined for this problem so that the design space is easily mapped and studied, as shown in Fig. 3. The shaded areas denote regions of infeasibility. For subspace optimization purposes, the design variables are allocated in a disjoint fashion in which CA_1 controls x_1 and CA_2 controls x_2. Also note that four different local optima exist in the design space (denoted by the stars); the optimal solution obtained is dependent on the starting point.

Results of Barnes's Problem

Starting from the design point indicated by the open circle in Fig. 3, $x = (40, 50)$, the CSSO algorithm drove the design into the region of the local optimum at $\boldsymbol{x} \approx (49.5, 19.5)$. At this particular optimum, true convergence of an approximate optimization algorithm is difficult to achieve because it lies on only one constraint so that it is not fully bounded ($n_{g_{\text{active}}} < n_x$). Indeed, it can be seen in Figs. 4 and 5 that the ability to converge (based on $\Delta \boldsymbol{x}$ and Δf) was highly dependent on the move-limit strategy employed for the coordination procedure. Figures 4 and 5 show the convergence histories of the objective, the trust region ratio, and the design variables and move-limits for each of the strategies tested.

Inspection of the plots in Figs. 4 and 5 shows that almost all move-limit strategies were able to allow CSSO to find the region of the local optimum rather quickly (within 5–6 iterations). This is not unexpected because this is only a two-dimensional problem, and the path to this optimum is unobstructed. However, the ability to actually converge to the optimum was only truly realized by the trust region approaches (DTR and TRAM) and the Thomas et al.[7] strategy. In comparing the trust region approaches, the DTR method converged in 11 iterations, whereas TRAM converged in 10. Note, in the move-limit convergence plots, that these trust region strategies were able to recognize that a local minimum had been passed, at which point the point-rejection scheme reduced the move limits significantly, allowing the algorithm to converge. The plot of the trust region ratio ρ convergence highlights the ability of TRAM to maintain the acceptable value (0.5) of the trust region ratio throughout most of the design process, whereas most of the other strategies exhibit high oscillations with respect to this quantity; the direct trust region approach required an extra point rejection. Thus, because TRAM allows for a continuous distribution of the move-limit adjustment factor between 0.5 and 2, more flexibility is allowed for in the control of the trust region ratio in subsequent iterations.

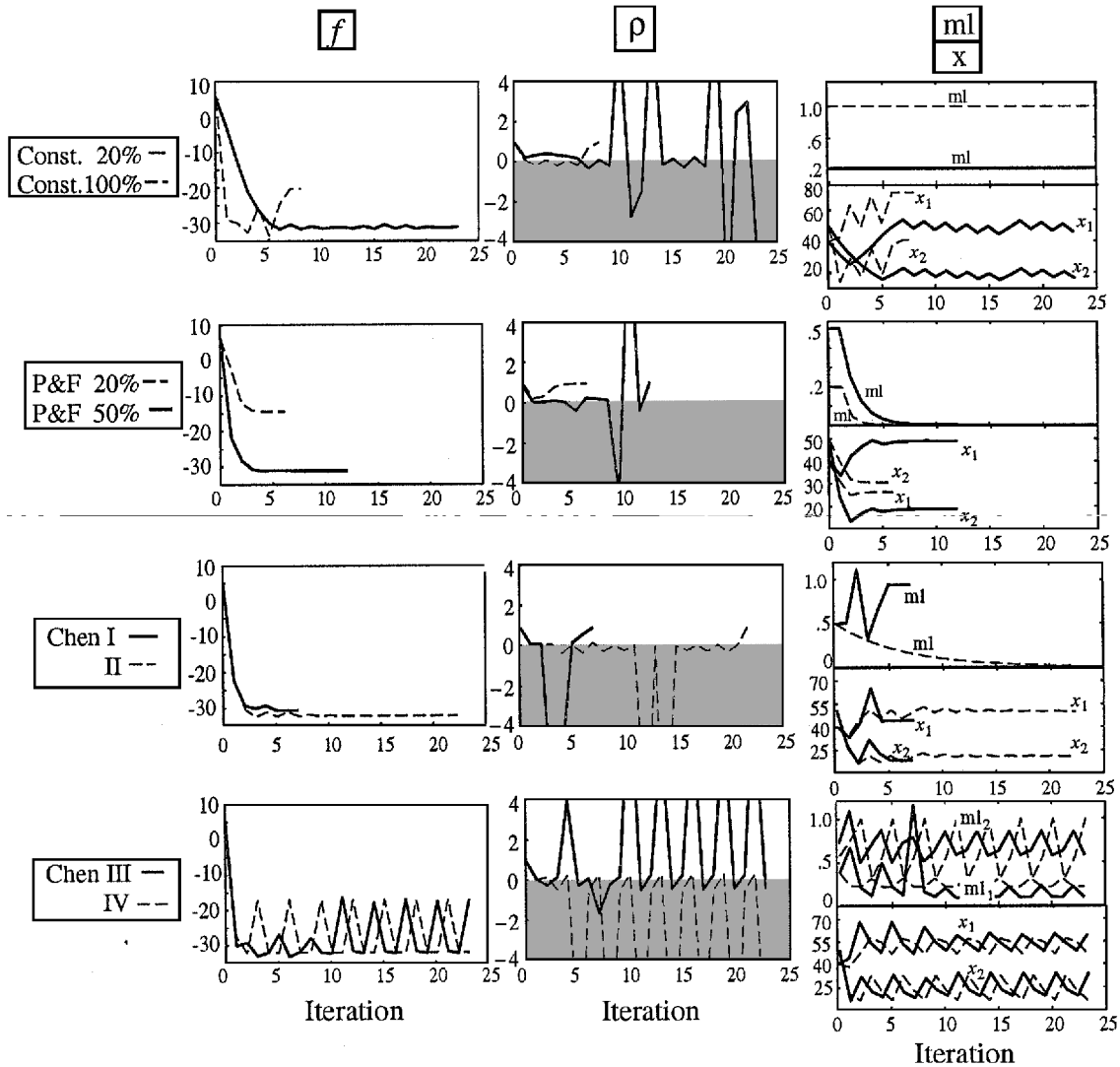

Fig. 5 Convergence plots for Barnes's[10] problem: constant/Chen[9] strategies.

The Thomas et al.[7] strategy was the only other strategy that eventually converged the algorithm, but only after 23 iterations. This is because it is in part based on the history of the design movement, and the detection of successive constraint violations allows for move-limit reduction. However, the inability to detect approximation error or a local optimum hinders this process. Implementation of the Bloebaum et al.[8] strategy on this two-dimensional problem makes evident that for only two design variables this strategy reduces to the assignment of constant move limits. Because of the way in which e^u and e^l are defined as linear functions of σ_e in the strategy formulation described earlier, it turns out that the linear distribution results in move limits of 8% and 42% for the two design variables (when $ml^u = 50\%$ and $ml^l = 0.1\%$) irrespective of e_i. The only difference from one iteration to the next is that the design variable with the higher e is assigned the 42% move limits. The result is a long, narrow rectangular region in which movement is allowed so that the design movement in that iteration is dominated by the design variable associated with the major axis.

The constant strategies exhibit oscillation about the minimum as expected, and manual termination was required. From a practical standpoint, manual termination may seem to be a viable alternative; however, this results in very subjective criteria that are not based on mathematical convergence criteria and cannot be accepted for general application. As a consequence, the process may be prematurely terminated when greater gains could be realized. Actually, when extremely large constant move limits are used (100%), the design jumps to a different local minimum at $\boldsymbol{x} \approx (71.5, 40.9)$ and converges because it is completely restricted by two active constraints so that error in the objective approximation is not a factor. The constant reduction strategies converge to a nonoptimal point because the move limits are reduced to zero after enough iterations. In fact, as discussed earlier, the progress of algorithms using constant reduction strategies is highly dependent on the initial move limits; in this problem, the initial move limits of 20% did not allow the design to reach the local optimum. The methods of Chen[9] exhibit extreme oscillation and only his method II converges to a point that is very near the optimum after 22 iterations because it reverts to a constant reduction scheme. His method I actually achieves false convergence at a suboptimal point (where the constraints intersect) because of poor objective approximations that it cannot account for.

The results just discussed emphasize the excellent convergence characteristics of move-limit management schemes based on the trust region approach. Specific to the TRAM method is a more refined ability to control the error in the approximations used to model the design space. Figure 6 shows the amount of error in the approximations of the objective function and constraints at the end of each iteration. The error in the approximation for g_3 was almost nonexistent and, thus, is not shown here. Although the TRAM method compares favorably to all methods in regard to the error, the most important comparison is with the direct trust region approach because only these two were shown to converge in an efficient manner. Note that the error in the approximations when the TRAM strategy is used is less than when the direct trust region is used in all but one iteration (iteration 7). The error allowed by the other strategies is often quite high and is not driven to zero (at convergence) as it is in the TRAM strategy. The inability to account for this error is

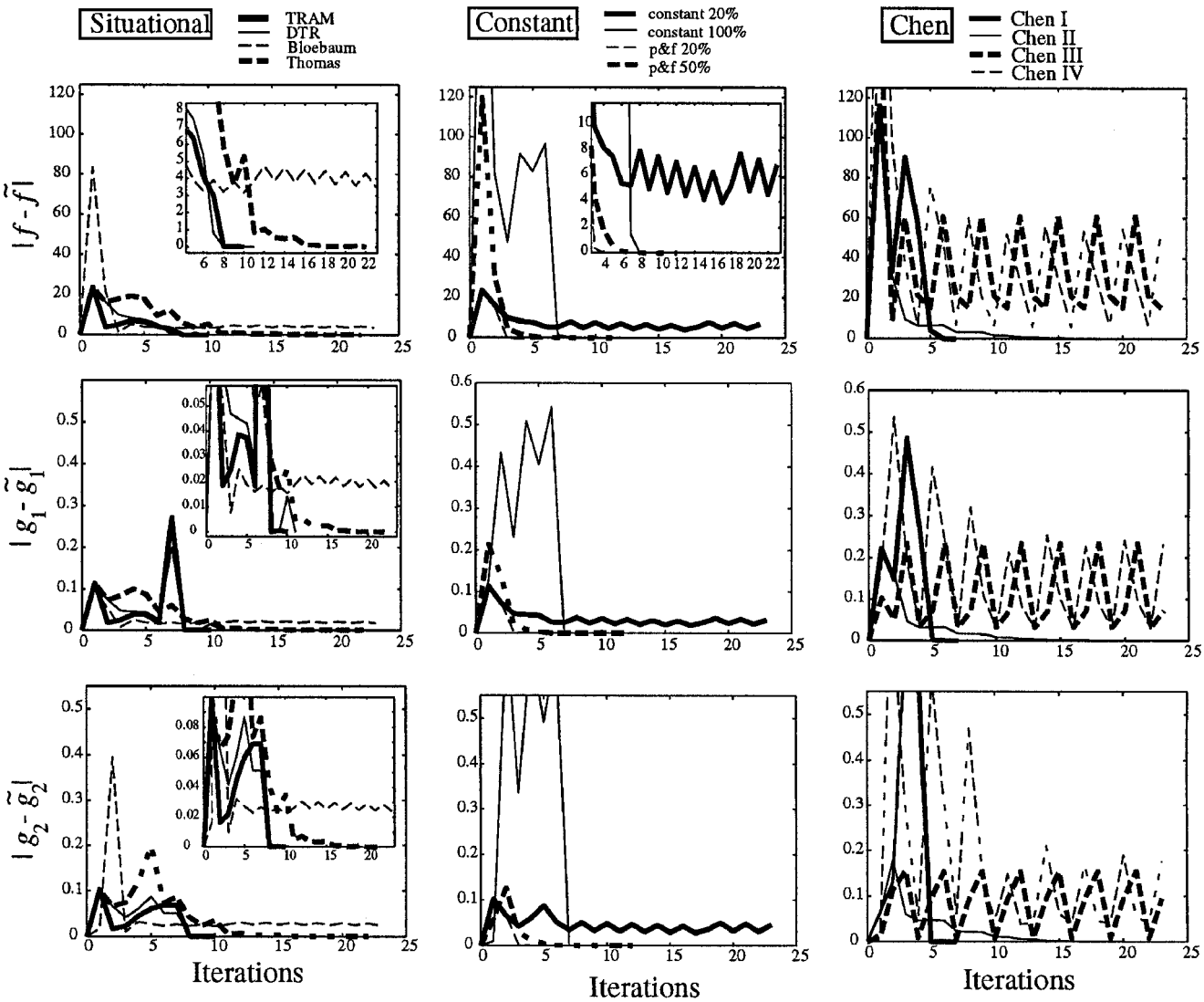

Fig. 6 Error plots for Barnes's[10] problem.

the main reason that these other strategies are not able to truly converge to the optimum. Thus, accounting for approximation error is a crucial component of the TRAM strategy to guarantee convergence of the algorithm.

High-Performance Low-Cost Structural Design

The preceding implementations have been used to compare the relative effectiveness of the TRAM strategy and the other move-limit strategies on small problems (few design variables and constraints) for easy interpretation and illustration of results. Because application is eventually intended for the design of large, complex systems, it was desired to implement the strategies on larger MDO test problems.

The first of these is the design of a high-performance low-cost (HPLC) structure as shown in Fig. 7a. This problem was first introduced in Ref. 2. The objective is to find the size and shape of the 10-bar truss such that the weight W_{tot} of the structure is a minimum (low cost), while the loads P_{1-4} it is capable of sustaining and the payload M_{1-4} it carries are a maximum (high performance). This multiobjective problem is formulated as a single objective problem by defining a cost-performance index (CPI) that includes each of the objectives as shown in Eq. (1). The coefficients w_{1-3} in the CPI are introduced to scale the separate components so that no one component dominates the others in driving the optimization. The design is subject to minimum payload and load requirements, as well as yield stress and first natural frequency constraints. A total of 13 inequality constraints exists for this problem. Equation (1) lists the problem in standard form.

Minimize:

$$\text{CPI} = w_1 W_{\text{tot}} + \left(w_2 \Big/ \sum P_i\right) + \left(w_3 \Big/ \sum M_i\right)$$

Subject to:

$$g_1 = 1 - \frac{(M_{\text{tot}})_{\min}}{\sum M_i} \geq 0.0, \qquad g_2 = 1 - \frac{(P_{\text{tot}})_{\min}}{\sum P_i} \geq 0.0$$

$$g_3 = 1 - \frac{\omega_{1,\min}}{\omega_1} \geq 0.0, \qquad g_{4-13} = 1 - \frac{|\sigma_{1-10}|}{\sigma_{\text{yield}}} \geq 0.0$$

$$x_i^{(l)} \leq x_i \leq x_i^{(u)}$$

where $w_1 = 0.003$, $w_2 = 10^6$, $w_3 = 3.5 \times 10^6$, $(M_{\text{tot}})_{\min} = 5000\,\text{lb}$, $(P_{\text{tot}})_{\min} = 100{,}000\,\text{lb}$, $\omega_{1,\min} = 2.0\,\text{Hz}$, and $\sigma_{\text{yield}} = 14{,}000\,\text{psi}$. The loads P_{1-4} applied to the structure are defined to be a function of the lengths of the bays, L_{1-3}, and the payload masses M_{1-4} placed on the structure as

$$P_i = \sum_{k=1}^{3} a_k^i \left(\frac{L_k}{L_{\text{ref}}}\right)^{b_k^i} + \sum_{j=1}^{4} c_j^i \left(\frac{M_j}{M_{\text{ref}}}\right)^{d_j^i} \tag{1}$$

where $L_{\text{ref}} = 60\,\text{ft}$ and $M_{\text{ref}} = 1250\,\text{lb}$. This is similar to an aeroelastic structure in which the loads incurred are dependent on the size and shape of the structure. The coefficients (a, b, c, and d) in Eq. (1) are listed in Ref. 2.

The design vector in this problem is composed of the length of the rectangular first bay, L_1, and the top and bottom lengths of the outer bay, L_2 and L_3, the masses (payload) placed on all of the unconstrained nodes, M_{1-4}, and the areas of the truss members, A_{1-10}. Thus, this is both a sizing and topology optimization problem. In all, 17 design variables are defined for this problem.

The design team is decomposed into three disciplines (CAs) as shown in Fig. 7b. The configuration design subspace (CA_c) calculates the weight of the structure, W_{tot}, and the loads P_i applied to the structure. CA_c is allocated the design variables L_i and M_i for subspace optimization with the areas (A_i^0) fixed. The loads P_i are used by the structural design group (CA_s) to calculate the deflections δ_i and stresses σ_i of the truss members. This analysis is performed using a finite element code in which the truss members are modeled as rod elements capable of sustaining axial loads only. CA_s is allocated the design variables A_i for subspace optimization with the lengths L_i^0 and masses M_i^0 fixed. A dynamic response design subspace (CA_d) calculates the natural frequencies ω_i of the structure, again through use of a finite element code. CA_d is allocated the design variables M_i for subspace optimization with the lengths L_i^0 and areas A_i^0 fixed. Note that the masses M_i are shared by the configuration and dynamics subspaces. In all, 35 states are calculated among the three CAs for this design problem. Note that simple coupling exists in this problem because the loads P_i output as state information from CA_c are required as input to the finite element code of CA_s in calculations of deflections and stresses.

By executing this optimization problem using the generalized reduced gradient method (OPT3.2)[11] from a number of starting points and with different tolerance parameters, a region of locally optimal designs was identified. It was found that different combinations of masses (x_4–x_7) and member areas (x_8–x_{17}) were able to achieve relatively the same minimum objective function of $f \approx 11.10$ for a given optimal configuration (x_1–x_3). Despite the different designs, all of the same constraints are active and are of relatively the same values. It is believed that the ability to achieve the same objective with different mass and area combinations is possible due to the definition of the loads in Eq. (1). These observations are in agreement with those reported in Ref. 2, although a lower objective was obtained here, possibly due to improvements in the optimizer OPT3.2 as described in Ref. 3. In Ref. 2, the configuration did not progress to that defined by the bounds on the bay lengths as was found in the present study. It was realized that from most starting points, the design could rather easily progress to the vicinity of the optimal design regarding the objective function. The difficulty in attaining further improvement is most likely because a number of constraints become active during the optimization process, and traveling along these constraints can be quite difficult. Both scaling and tolerances become major issues in the region of the optimum, so that fine tuning the design to truly converge to an optimum is a laborious process.

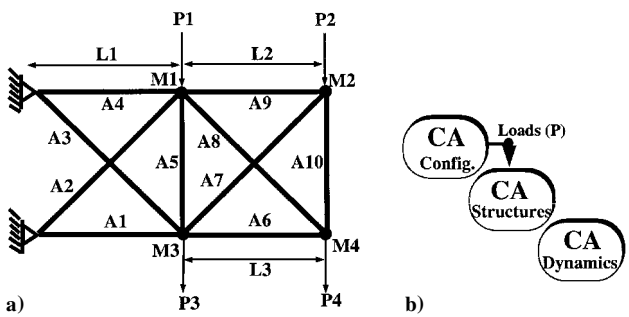

Fig. 7 HPLC structure.

Results for the HPLC Structural Design Problem

Based on the results obtained from implementation on the first problem and others,[3] it was decided that testing the constant reduction strategies was unnecessary because they simply reduce the move limits to zero and are highly dependent on the starting point. Furthermore, the ability to reach the region of the optimum as earlier described was independent of the starting point[3]; thus, a single starting point was used in this study and was chosen to be $L_{1-3} = 360$ ft, $M_{1-4} = 1500$ lb, and $A_{1-10} = 12$ in.^{2}, the same initial design used for the study in Ref. 2. An in-depth discussion of the history of the design from this starting point as directed by the CSSO algorithm is given in Ref. 2. Here the discussion will focus on the characteristics relevant to the present move-limit study.

Because of the number of constraints and design variables in this problem, it is not possible to plot all quantities in a concise fashion. However, a good understanding of the effectiveness and efficiency of the various move-limit strategies can be obtained by plotting only a few of the more important quantities. The convergence histories of the objective, f, and the TRAM merit function Φ are plotted together for each method to highlight the incurrence of penalty due to constraint violation. The trust region ratio ρ is plotted to give an indication of the amount of error in the approximations allowed by each method. Finally, the history of the move limits is plotted to illustrate the move-limit recommendations of each strategy.

To illustrate the need for an adaptive strategy, the results of applying constant percentage move limits to this problem are shown in Fig. 8. Note that when 30% move limits are applied, the region of the optimum is found rather quickly (9 iterations) with one intermediate design iterate quite violated (as indicated in plots of both Φ and ρ).

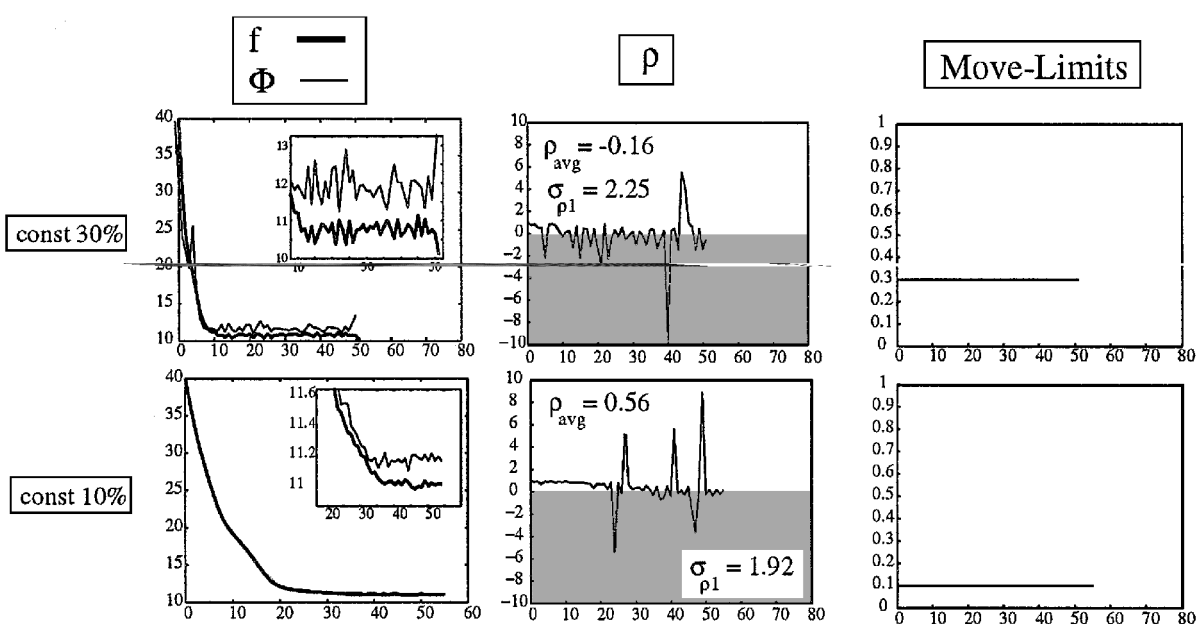

Fig. 8 HPLC structural design results with constant move limits.

However, once the region is found, the 30% move limits do not allow the algorithm to converge on an optimal design due to the amount of error in the approximations. Again, this error and the associated constraint violations are evident in the plots of both Φ and ρ. In the plot of Φ, it is evident that the amount of constraint violation is so great that the objective is penalized by approximately 10% of its value. In the plot of ρ, practically every other iterate has a negative ρ value (the shaded region), indicating that very poor approximations were used. The algorithm was manually terminated because convergence could not be achieved. When the constant percentage was reduced to 10%, it took longer to find the region of the optimum (19 iterations), but it did so with practically no approximation error ($\rho \approx 1$). However, a similar difficulty in the ability to converge was observed, although the constraint violations and approximation error were not as great. Still, it was necessary to manually terminate the algorithm because convergence was not obtained.

Based on the constant move-limit results, the initial move limits for all other strategies were set at 10%. The results for the situational strategies are shown in Fig. 9. As expected, the two trust-region-based strategies are able to adjust the move limits such that true convergence can be obtained. As observed in the smaller test problems, the TRAM strategy is able to allow less approximation error than the DTR strategy and, thus, finds the region of the optimum faster (fewer rejected points) and converges faster (35 iterations as opposed to 49). Although the average ρ value is greater for the DTR method, this is deceiving because many positive spikes are observed in the ρ value. Thus, there are more instances in which favorable approximation error exists in the DTR method, i.e., the TRAM merit function decreased more than estimated, which increases the average. Yet, any deviation from $\rho = 1$ indicates approximation error, so the standard deviation of ρ from 1 ($\sigma_{\rho 1}$) is a more accurate indication of the allowed error. This value is significantly lower for the TRAM strategy, indicating smaller allowed error in the approximations. Note the eventual reduction of the move limits to very small values in each of these trust region strategies as the minimum is detected through the trust region ratio.

The Bloebaum et al.[8] strategy finds the region of the optimum rather efficiently in approximately the same number of iterations as the trust region strategies with little approximation error. However, as observed in the constant percentage strategies, this strategy is not able to account for approximation error and, thus, cannot adjust the move limits accordingly to converge on the optimum. A great amount of approximation error is allowed, as observed in the plot of ρ, and the move-limit settings are erratic (exemplified by the move limits for x_1 and x_{17}) due to their strict dependency on the constraint and objective gradients. The strategy proposed by Thomas et al.,[7] which was found to work for the smaller problems to a certain extent, was not able to even reach the region of the optimum let alone converge to an optimal point. This is most likely explained by the way the strategy reduces the move limits when the constraint violation increases in consecutive iterations, a common occurrence in this problem because numerous constraints are active. It can be seen that the move limit for x_1 quickly reduces to a very small value, as was observed for the other design variables. Thus, the design is not even given a chance to reach the region of the optimum.

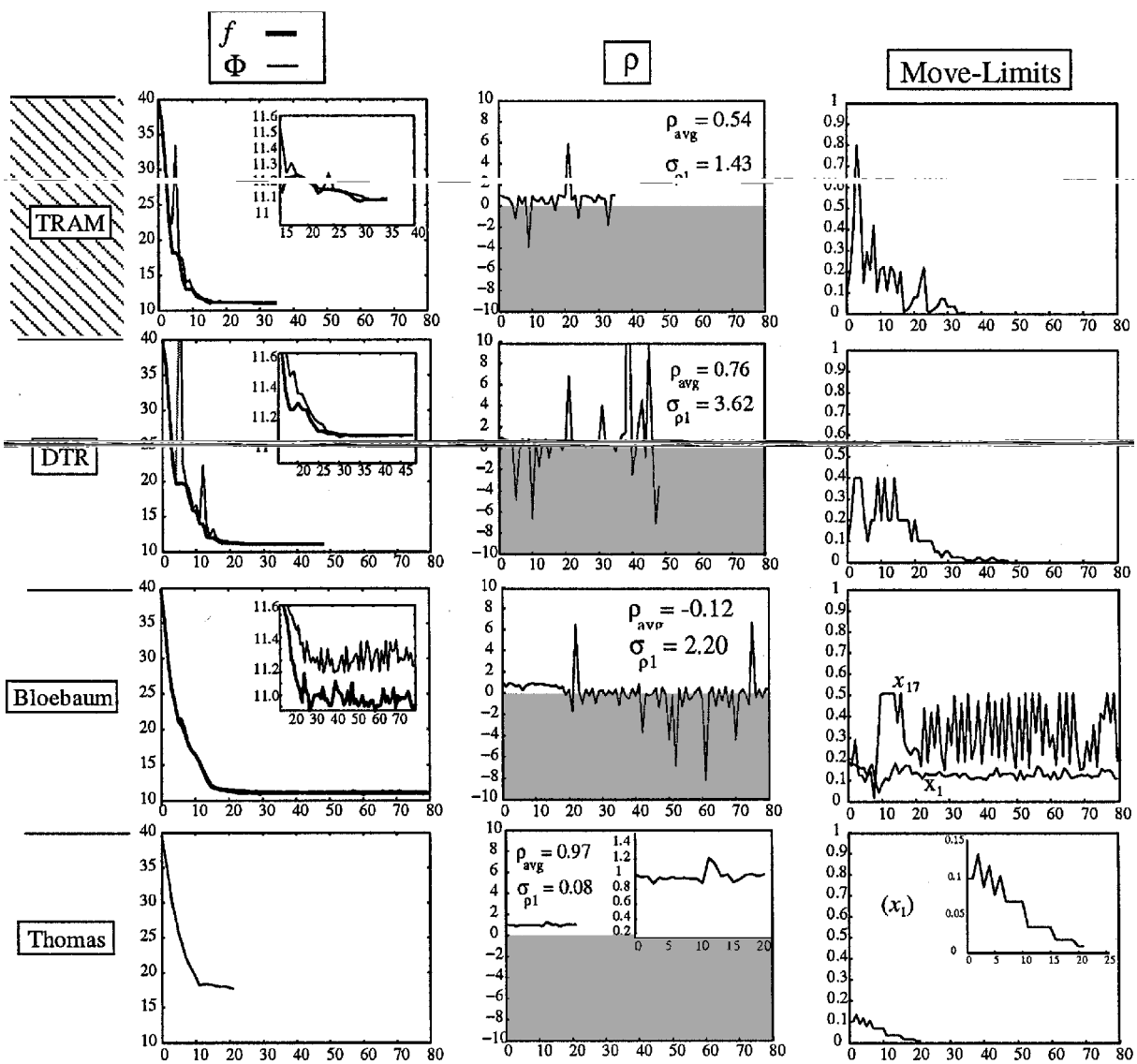

Fig. 9 HPLC structural design results with situational move limits.

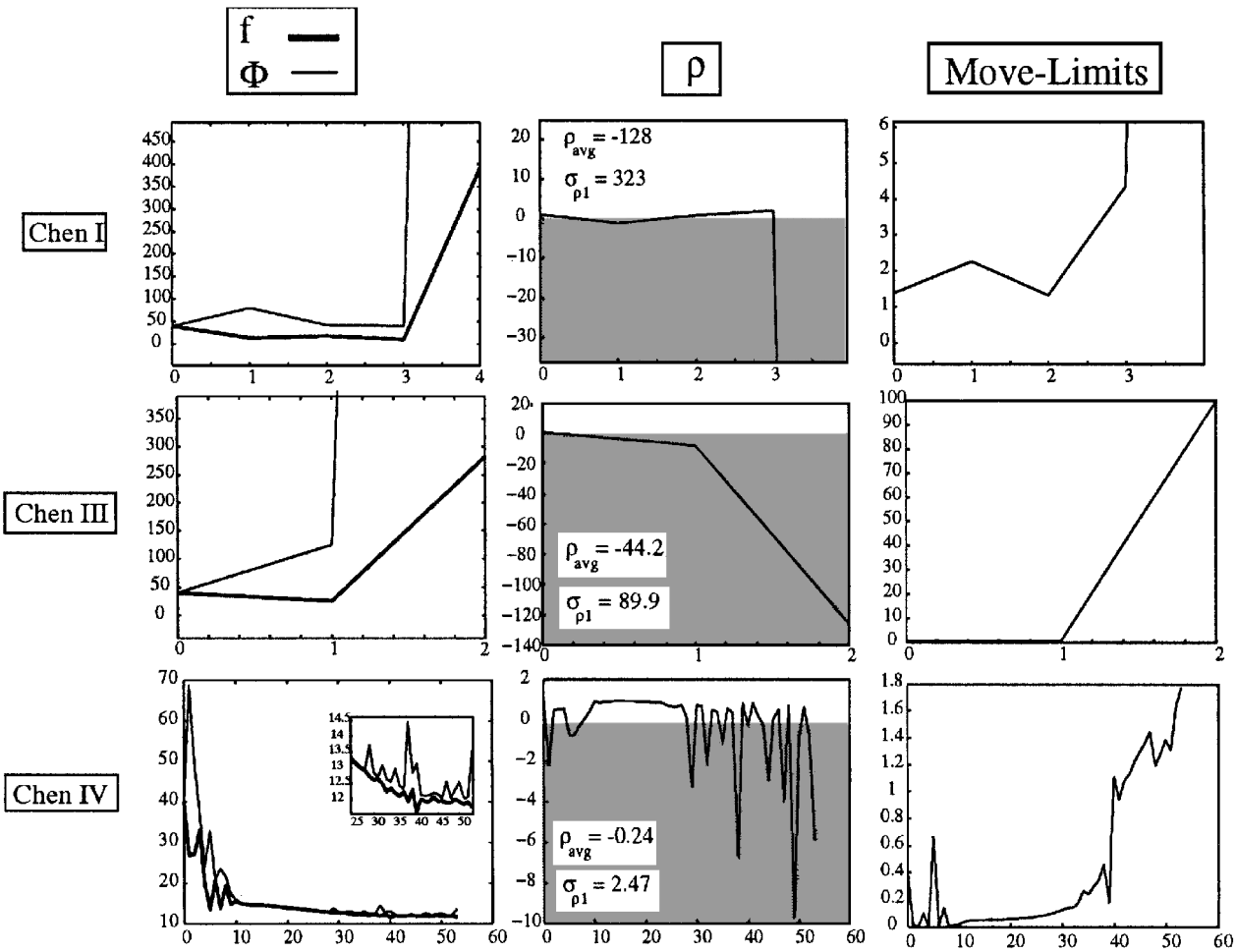

Fig. 10 HPLC structural design results with Chen[9] move limits.

The results of applying Chen's[9] strategies to this problem are shown in Fig. 10. The Chen II strategy was not implemented because it employs constant reduction beyond the first iteration. In Chen's strategies I and III, extremely large move limits are set based on the distance to the farthest constraint, as Chen suggests. The fault in this reasoning is apparent because the resulting approximation error drives the design to a very infeasible region, increased objective value (because the objective is not considered by these strategies), and eventual divergence. As recognized in the earlier test problems, Chen's strategy IV is the most effective of the group because it includes the objective function. It is able to progress to near the region of the minimum, although a significant amount of error is incurred because the approximation is not accounted for. It does not quite reach the minimum, but instead oscillates around $f \approx 12$ with a significant amount of error, as indicated in the plot of ρ. Manual termination was required.

Overall, the results obtained using the various move-limit strategies on this larger test problem are consistent with those found for the smaller problems. The TRAM strategy effectively reaches a converged optimal solution and controls the amount of error incurred in doing so. The need for such an adaptive move-limit management strategy is evident when the results of applying constant move limits are observed. Likewise, other strategies that are based on gradients or history alone are not able to accommodate the effect that approximation error has on proper move-limit settings. The trust region approach is the most effective method, and the flexibility offered by the TRAM implementation allows for more refined move-limit adjustment than DTR, resulting in less approximation error and faster convergence.

HPLC Structural Design: Formulation II

It has been mentioned that the TRAM move-limit strategy is not specific to CSSO, but can be applied to any approximate optimization algorithm provided that the necessary information is available. To illustrate this assertion, the TRAM strategy was tested for use in the successive linear approximations (SLA) algorithm on a reformulated version of the HPLC structural design problem already described. In HPLC II (Fig. 11), Euler buckling constraints are enforced for all members in compression, adding the 10 new constraints

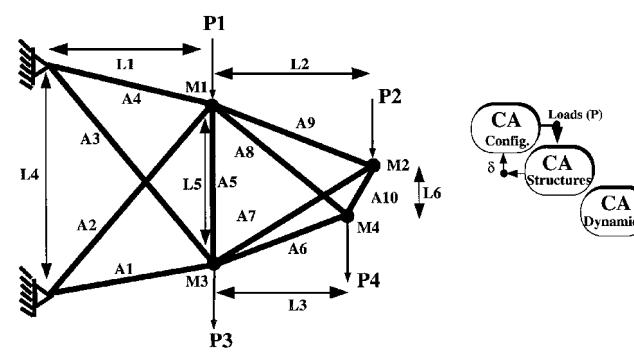

Fig. 11 HPLC structural design problem: formulation II.

$$g_{j+13} = \begin{cases} 1 & \text{if} \quad \sigma_j \geq 0 \\ 1 - \dfrac{|\sigma_j|}{\sigma_{j,\mathrm{cr}}} & \text{if} \quad 0 > \sigma_j > -\sigma_{j,\mathrm{cr}} \\ -1 + \dfrac{\sigma_{j,\mathrm{cr}}}{|\sigma_j|} & \text{if} \quad 0 > -\sigma_{j,\mathrm{cr}} > \sigma_j \end{cases}$$

$$\sigma_{j,\mathrm{cr}} = \frac{\pi E_j A_j}{4L_j^2}, \qquad j = 1, \ldots, 10 \quad (2)$$

to the optimization problem listed in Eq. (1). In addition, greater flexibility is allowed in the configuration of the structure by including the heights of the bays as three additional design variables

$$x_{18-20} \equiv L_{4-6} \tag{3}$$

Thus, it can be seen that the dimensionality of the problem is increased and the feasible region of the design space is reduced. A final modification to the problem introduced complex coupling in the system analysis by feeding the nodal deflections calculated by the structures CA back to the configuration CA to calculate updated load values. To account for this, the loads are redefined as

$$P_i = \sum_{k=1}^{3} a_k^i \left(\frac{L_k}{L_{\text{ref}}}\right)^{b_k^i} + \sum_{j=1}^{4} c_j^i \left(\frac{M_j}{M_{\text{ref}}}\right)^{d_j^i} - e_i \delta_{y,i} \tag{4}$$

The coefficients (a, b, c, d, and e) for this definition of the loads are listed in Ref. 3. This dependency provides a truer simulation of aeroelastic coupling than the original formulation.

Results for HPLC Structural Design II

The addition of the buckling constraints in this problem creates a more formidable challenge for the optimization process because members 1, 2, 6, and 7 are all in compression and all of the associated buckling constraints are active at the optimum. As noticed for the original formulation, the design space of formulation II contains a region of optima, a location in the design space in which different local optima can be reached with only slight adjustments (or interchanging) of a number of the design variable values. Because of this, the tolerance used to specify constraint activity plays an important role in the optimization process; if slight violation can be accepted in the region of optima, it is possible to bypass those local optima with greater objective function values. A NAND[12] implementation on the entire system in OPT3.2 (which tolerates no constraint violation) converged to an optimum with an objective function value of 19.30, but this was achieved only after 14 restarts of the algorithm to allow for scale factor updates (essentially the optimizer walked along the constraints). As presented in the following discussion, SLA implementations, in which the move limits were governed by the trust region approaches (TRAM and DTR), were able to obtain lower objectives (19.0–19.1) because slight constraint violations are tolerated. Results of the SLA implementations with various move-limit strategies are discussed next.

The results for this test problem are similar to those observed for formulation I except that constraints are reached faster so that adjustments to the move limits are critical earlier in the process. Because of this, the initial move limits were again set at the conservative level of 10%. It can be seen in Fig. 12 that constant move limits (even as low as 10%) do not allow convergence because constraint violation (from approximation error) is not accounted for. A great amount of error is allowed in the approximations in the region of the optimum, as evidenced in the standard deviation of ρ from 1 ($\sigma_{\rho 1} = 18.64$). The algorithm was necessarily manually terminated.

The trust region strategies were the only methods to truly converge to an optimal design based on their ability to reduce the move limits around the optimum. Both TRAM and DTR initially increase the move limits based on good ρ values (low approximation error and no infeasibility). The flexibility of the TRAM adjustment factor setting allows less approximation error than DTR overall (see ρ_{av} and $\sigma_{\rho 1}$ values; Fig. 12). Eventually each of these strategies reduces the move limits to very small values (due to the detection of constraint violation) until the optimum is converged on. Note in the plots of f and Φ that convergence to the optimum coincides with Φ converging to f, i.e., no constraint violation. Again, the flexible TRAM adjustment factor leads to convergence in 10 fewer iterations than DTR (a 20% improvement).

Because the Bloebaum et al.[8] strategy is merely based on objective and constraint gradients, it does not detect approximation error or constraint violation. Thus, great amounts of these quantities are observed in the plots of ρ and Φ. Also note the erratic move-limit settings, which are often observed with this strategy. The Thomas et al.[7] strategy detects constraint violation almost immediately and continually reduces the move limits from the outset. Some increases are observed (as in x_{13}) based on variables hitting their bounds in consecutive iterations, but these, too, are eventually reduced to very small values. Whereas this results in negligible approximation error (as seen in the plot of ρ), it also leads to premature convergence ($f = 21.7$) due to total restriction of design variable movement. Only Chen's[9] strategy IV was tested on this problem based on previous observations of the ineffectiveness of the first three Chen strategies on other problems. This strategy exhibits performance similar to the Bloebaum et al.[8] strategy, a result which might be expected due to the similarity of the methods. It is able to find the region of the optimum rather quickly, but significant error is allowed in the approximations (see ρ plot) and a great amount of infeasibility is encountered (see Φ plot). The move-limit settings are also quite erratic. Interestingly enough, however, the algorithm just happens to jump to a point very near the optimum at iteration 68 and can almost be considered converged. Yet, because the strategy has no mechanism for detecting the optimum and making the necessary move-limit reductions, the algorithm never truly converges to the tolerance of 0.001% set for both Δf and $\Delta \boldsymbol{x}$.

Design of an Autonomous Hovercraft (AHC)

The final problem used to compare the TRAM strategy with existing move-limit strategies is the design of an autonomous hovercraft (AHC), as shown in Fig. 13. This problem was first presented in Ref. 13, and a thorough description can be found in Ref. 14. The physical system consists of an engine, rotor (two rectangular lifting surfaces attached to the ends of a hollow circular shaft), and payload. The system is to operate such that the motor speed (revolutions per minute) provides a thrust-to-weight ratio of one, imposing a hover condition.

The system analysis is composed of four contributing analyses, three of which are complexly coupled as shown by the dependency diagram in Fig. 13. The aerodynamics CA (CA_a) calculates the aerodynamic loads on the lifting surfaces and approximates the distributed drag force along the rod, estimating the induced velocity at the lifting surfaces as a function of the thrust. CA_a requires the torsional deformation of the shaft, θ_d, and the motor revolutions per minute and thrust as inputs. The shaft deformation is supplied from a structures CA (CA_s), which also calculates the axial and shear stresses of the rod at the hub and the deflection of the lifting surfaces. All of these quantities are functions of the loads computed by CA_a; thus, these CAs are subject to static aeroelastic coupling. A propulsion/performance CA (CA_p) calculates the thrust and torque necessary to spin the rotor based on the loads supplied from CA_a. It is in this CA that the hover condition is imposed by driving the design to a thrust-to-weight ratio of one. In Ref. 14, this is accomplished by heuristically updating the revolutions per minute of the motor, which is used by CA_a to recalculate aerodynamic loads. In the present study, a reformulated version of CA_p calculates revolutions per minute as a state. This is done by explicitly enforcing that total weight equals thrust, determining the weight of the motor to achieve this total weight, and calculating the power and resulting revolutions per minute available from that size motor (based on an empirical relation). The coupling of CA_p with CA_a is instigated by the newly calculated motor revolutions per minute. A fourth CA, structural dynamics (CA_d), calculates the first natural frequencies of the rotor in bending and torsion. CA_d is completely uncoupled from the other CAs as it requires no states as inputs.

Minimize:

$$f(\boldsymbol{x}) = W_{\text{empty}} = W_{\text{wing}} + W_{\text{rod}} + W_{\text{fuel}} + W_{\text{motor}}$$

Subject to:

$$g_1 = 1 - \frac{\sigma_N}{\sigma_{\text{all}}} \geq 0.0, \qquad g_2 = 1 - \frac{\sigma_T}{\sigma_{\text{all}}} \geq 0.0$$

$$g_3 = \frac{\omega_b}{\text{k} \cdot \text{rpm}} - 1 \geq 0.0, \qquad g_4 = \frac{\omega_t}{k \cdot \text{rpm}} - 1 \geq 0.0$$

$$g_5 = \frac{M_{\text{tip}_{\text{all}}}}{M_{\text{tip}}} - 1 \geq 0.0, \qquad g_6 = \frac{E}{E_{\text{req}}} - 1 \geq 0.0$$

$$x_i^{(l)} \leq x_i \leq x_i^{(u)}$$

where $\sigma_{\text{all}} = 14{,}000$ psi, $k = 1.5$, $M_{\text{tip}_{\text{all}}} = 0.8$, and $E_{\text{req}} = 2$ h.

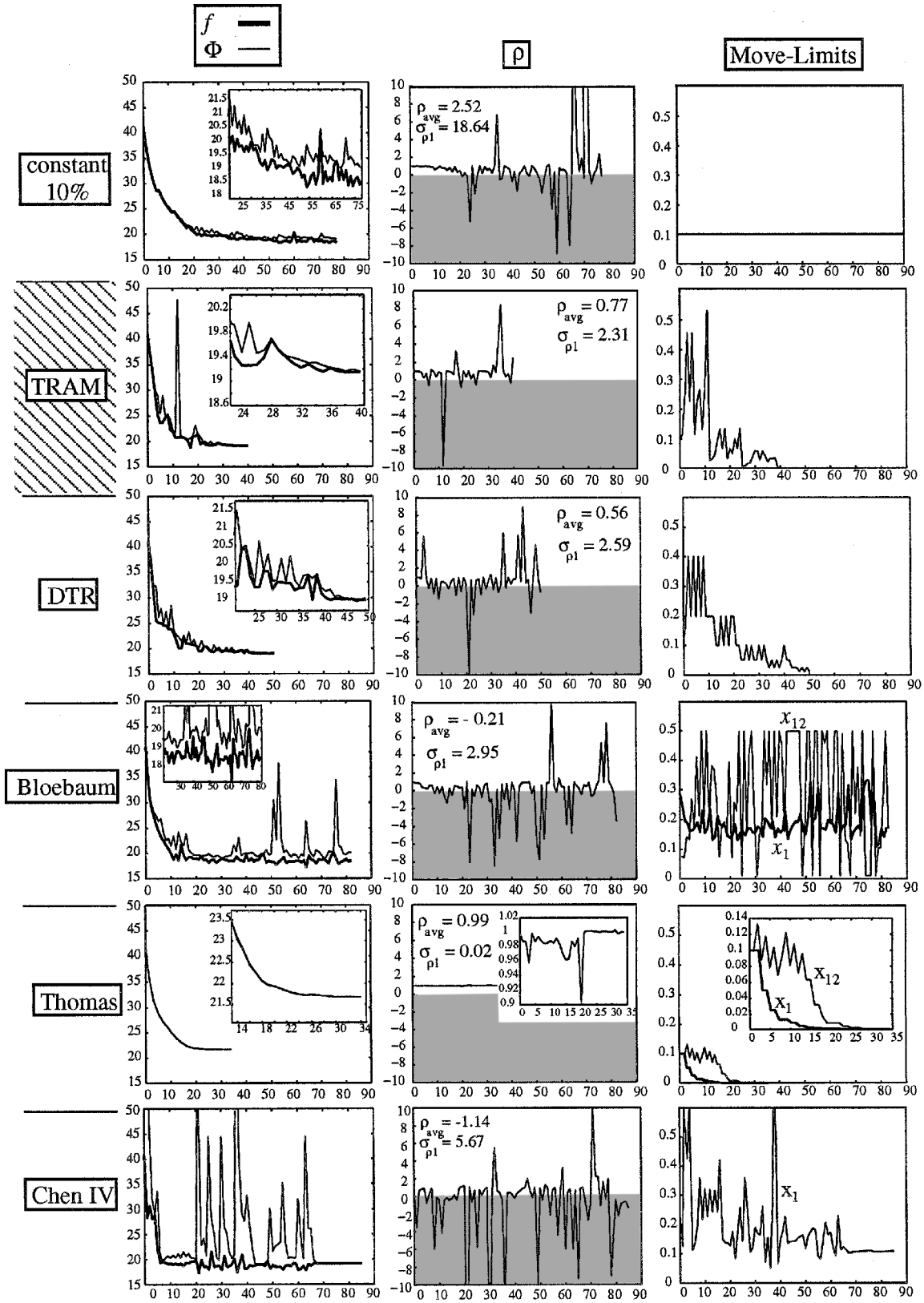

Fig. 12 HPLC II structural design convergence results.

In all, 11 design variables exist for this design problem describing the geometry of the rod and lifting surfaces and the amount of fuel. The CAs described calculate a total of 50 states as defined in this formulation. Of the problems studied in this research, the AHC problem most closely simulates the difficulties encountered in MDO, an observation based primarily on the complexity of the system analysis. Newton's method was necessarily employed for robustness of the system analysis convergence.

The goal of the optimization of this system is to minimize the empty weight of the hovercraft subject to constraints on the axial and shear Von Mises stresses in the rod, the first natural frequencies of the rod (relative to the motor revolutions per minute), the Mach number at the tip, and the endurance of the hovercraft. The optimization problem was formulated earlier.

The global optimum for this problem was reported in Ref. 14 to be $W_{\text{empty}} = 67.9$ lb, for which 7 of the 11 design variables are at global

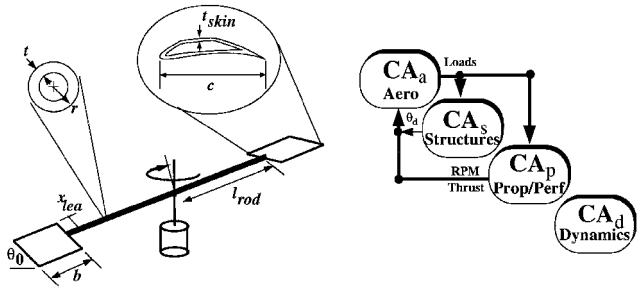

Fig. 13 Autonomous hovercraft system.

bounds and the endurance constraint g_6 is active. It was also reported in Ref. 14 that at the optimum the objective is very insensitive to two of the variables (camber and x/c) so that variations in these values can be expected of the optimal design obtained by CSSO.

Results for the AHC Design Problem

The optimization of this problem was carried out via the CSSO algorithm from a design that is highly infeasible with respect to the endurance constraint g_6. It was discovered that from this point the path to the optimum includes a search for a design that is feasible, improvement of the design in feasible space, and reencountering of g_6, which is traveled along to the optimum. The convergence results for the various move-limit strategies are shown in Fig. 14.

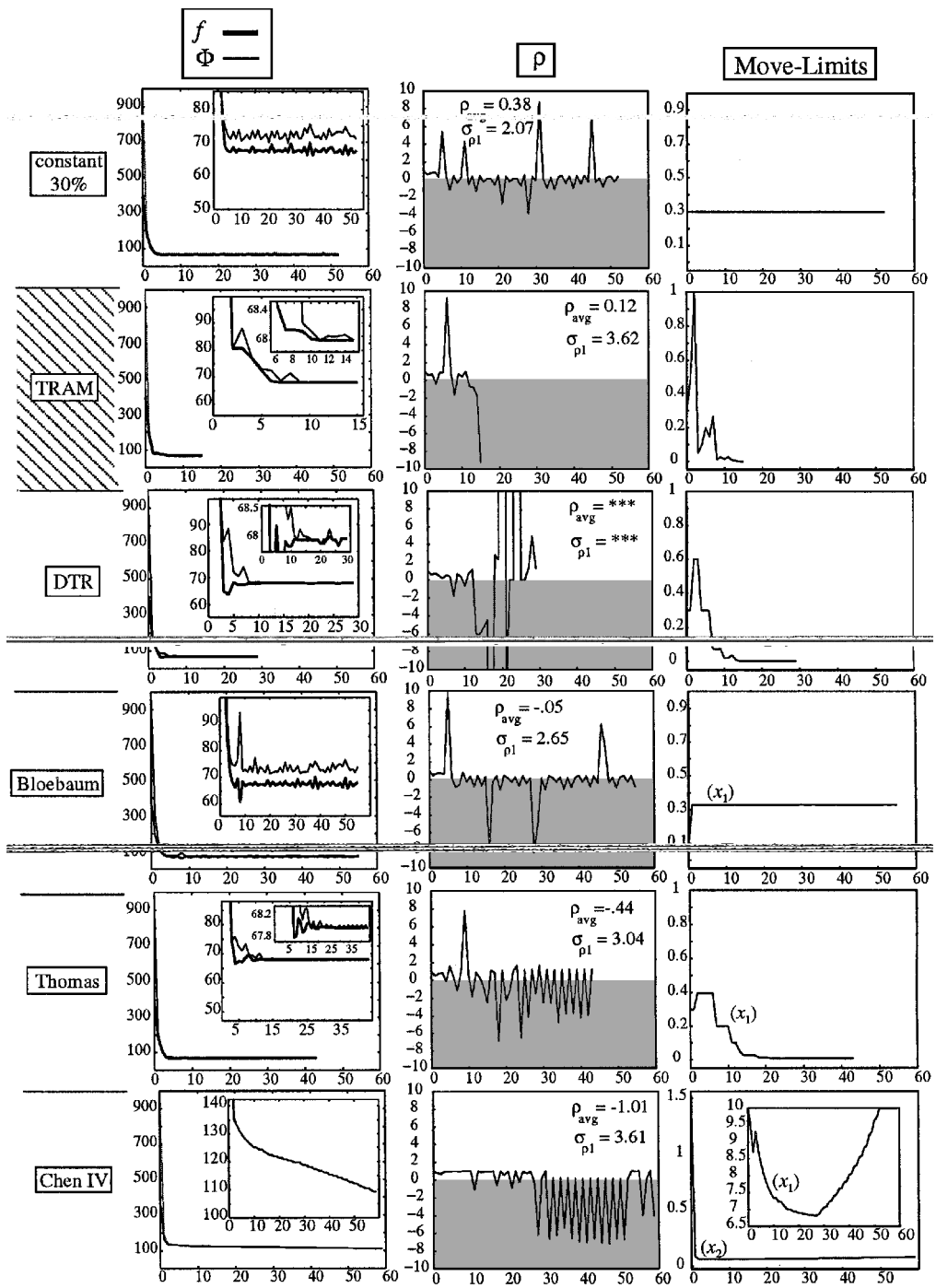

Fig. 14 Autonomous hovercraft convergence results.

As in the structural design problems, constant move limits (30%) are able to find the neighborhood of the optimum with little difficulty. But convergence cannot be achieved because infeasibility due to approximation error is not detected and accounted for. As seen in the history of ρ, almost every other iterate is what would be considered a failure because Φ actually increases ($\rho < 0$).

Detecting the gains that can be made with design variable freedom in the initial feasible search, the TRAM strategy increases the move limits in the first few iterations. When the design becomes feasible, the error in the objective function f dominates and the move limits are drastically reduced. After a few successful iterations (with move-limit increases), the endurance constraint is re-encountered and the move limits are again reduced. The minimum is detected along the constraint surface, and point rejection and the associated move-limit reduction lead to convergence in 15 iterations.

The high $\sigma_{\rho 1}$ value relative to the other strategies is somewhat misleading because the fewer design iterates are biased by the ρ values at convergence. It was observed that because ρ is actually undefined when $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)}$, the ρ value does not truly represent the amount of approximation error when $\Delta \boldsymbol{x}$ is very small, i.e., the ratio of Φ reduction is not representative. Yet, it still provides meaningful information for move-limit adjustment leading to convergence.

The DTR method converges in 29 iterations in a similar fashion to the TRAM strategy; although with more rigid move-limit adjustments greater penalty is incurred from constraint violation. The convergence plots of Fig. 14 show that DTR was very nearly converged after about 19 iterations, but the inflexible rejection scheme employed delayed convergence. In fact, it was necessary to modify the reduction factor to 0.25 instead of 0.5 (as is suggested in Ref. 15) to avoid cycling, which can occur with successive halvings and doublings. The earlier comments regarding ρ apply here also, where the effect of small $\Delta \boldsymbol{x}$ on ρ is even more predominant. In fact, the ρ values were so extreme that ρ_{av} and $\sigma_{\rho 1}$ were calculated to be ridiculously large values and are meaningless to present.

It is believed that some of the difficulties encountered in gaining true convergence (leading to the unpredictable ρ values) might arise from noise in the system analysis from the iterative convergence process. The states used to calculate f and $\boldsymbol{g}$ have some amount of error associated with them due to the tolerance used in the iterative procedure. This can have a significant effect on algorithm convergence as $\Delta \boldsymbol{x} \to 0$. Further study is required to investigate the validity of this assertion.

The results of the Bloebaum et al.[8] strategy are similar to those with constant 30% move limits, mainly because that is what the method calculates for many of the variables ($\approx$30% move limits in every iteration). Again, significant approximation error and infeasibility are encountered, and manual termination was required. As observed in some of the preceding problems, the Thomas et al.[7] strategy converges by detecting constraint violation, which eventually reduces the move limits to zero around the optimum. However, a total of 43 iterations was required to achieve convergence because increase in constraint violation is the only means of reduction. The failure of Chen's[9] strategy IV on this problem highlights another oversight by the method. The variables that are calculated to have the largest influence (based on linear approximations) reach their global bounds before the optimum is reached. Without accounting for this, the strategy calculates small move limits for those variables with lower associated sensitivities and progress is very gradual therein. The optimum would have eventually been reached after approximately 180 iterations if the algorithm had continued at that rate.

IV. Discussion

Similar observations can be drawn from each of the implementation studies presented. Table 1 provides a comparison of the characteristics offered by the strategies tested. It is evident from the results that constant move limits and constant reduction strategies are not adequate to converge approximate optimization algorithms and that significant approximation error and infeasibility are present with the use of such methods. Likewise, strategies that rely merely on gradient information (Bloebaum et al.[8] and Chen[9]) do not account for approximation error (nonlinearity) and inherently lead to infeasibility and difficulties in convergence (cycling). It was observed that relying on the history of previous design changes alone (Thomas et al.[7]) can sometimes result in beneficial move-limit adjustment and eventual convergence. But this procedure can sometimes be overwhelmed by approximation error and has difficulty traveling along constraint surfaces because move limits are drastically reduced leading to premature convergence.

The trust region approach to move-limit adjustment has been shown to consistently converge approximate optimization algorithms, supporting the proof of convergence offered in Ref. 16. Strategies based on this approach account for the history of previous design changes, approximation error, and infeasibility. In situations when the move limits are overly restrictive and unnecessary, trust-region-based strategies can increase the move limits to allow for greater design improvement. The ability to detect the minimum (through negative ρ values) allows trust region methods to reduce the move limits to converge on the optimum. Note that the one desired characteristic not currently supported by the trust region strategies is the ability to set individual move limits; this is addressed in recommendations for future research offered in Ref. 3.

V. Summary

The TRAM strategy developed in this research provides a more flexible move-limit adjustment factor than direct trust region methods through the use of sensitivity information. Coupled with a new approximation-based point rejection scheme, use of the TRAM strategy results in less approximation error and leads to faster algorithm convergence in most cases. Table 2 lists the percent improvement realized using TRAM over the DTR approach for the

Table 1 Characteristics of the move-limit strategies tested

Strategy	Basis	Account for approximation error	Account for infeasibility	Bound hitting	Individual variable assignment	Detect minimum	Adjustment flexibility	Provably convergent
Constant move limits/ constant reduction	Intuition Experience Guesswork							
Bloebaum et al.[8]	Sensitivity				√		√	
Chen[9]	Sensitivity				√		√	
Thomas et al.[7]	History		√	√	For increases			
DTR	History Approximation error Feasibility	√	√	√		√		√
TRAM	History Approximation error Feasibility Sensitivity	√	√	√		√	√	√

Table 2 Improvement realized using TRAM over DTR

Test problem	% Improvement over DTR
Barnes's[10] problem	9
HPLC structure	27
HPLC structure II	20
Hovercraft	48

problems studied here. These results indicate that on average a 20% improvement in algorithm efficiency can be obtained using TRAM as compared to DTR. Further testing on other problems is required to verify this assertion. Overall, the development of the TRAM strategy provides improved efficiency in the execution of approximate optimization algorithms.

Acknowledgments

This multidisciplinary research effort was supported in part by the following grants and contracts: NASA Research Grant Award NAG-1-1561, National Science Foundation (NSF) Grant DMI93-08083, and NSF Grant DMI94-57179.

References

[1]Wujek, B. A., and Renaud, J. E., "New Adaptive Move-Limit Management Strategy for Approximate Optimization, Part 1," *AIAA Journal*, Vol. 36, No. 10, 1998, pp. 1911–1921; also AIAA Paper 98-1965, April 1998.

[2]Wujek, B. A., Renaud, J. E., Batill, S. M., and Brockman, J. B., "Concurrent Subspace Optimization Using Design Variable Sharing in a Distributed Computing Environment," *Proceedings of the 21st ASME Design Automation Conference*, Vol. 1, DE-Vol. 82, American Society of Mechanical Engineers, New York, 1995, pp. 181–188.

[3]Wujek, B. A., "Automation Enhancements in Multidisciplinary Design Optimization," Ph.D. Dissertation, Univ. of Notre Dame, Notre Dame, IN, July 1997.

[4]Fadel, G. M., Riley, M. F., and Barthelemy, J. F. M., "Two Point Exponential Approximation Method for Structural Optimization," *Structural Optimization*, Vol. 2, No. 2, 1990, pp. 117–124.

[5]Grignon, P., and Fadel, G. M., "Fuzzy Move Limit Evaluation in Structural Optimization," AIAA Paper 94-4281, Sept. 1994.

[6]Pourazady, M., and Fu, Z., "Integrated Approach to Structural Shape Optimization," *Computers and Structures*, Vol. 60, No. 2, 1996, pp. 279–289.

[7]Thomas, H. L., Vanderplaats, G. N., and Shyy, Y.-K., "A Study of Move Limit Adjustment Strategies in the Approximation Concepts Approach to Structural Synthesis," *Proceedings of the AIAA/USAF/NASA/OAI 4th Symposium on Multidisciplinary Analysis and Optimization*, AIAA, Washington, DC, 1992, pp. 507–512.

[8]Bloebaum, C. L., Hong, W., and Peck, A., "Improved Move Limit Strategy for Approximate Optimization," *Proceedings of the AIAA/USAF/NASA/ISSMO 5th Symposium on Multidisciplinary Analysis and Optimization*, AIAA, Washington, DC, 1994, pp. 843–850 (AIAA Paper 94-4337).

[9]Chen, T.-Y., "Calculation of the Move Limits for the Sequential Linear Programming Method," *International Journal for Numerical Methods in Engineering*, Vol. 36, 1993, pp. 2661–2679.

[10]Barnes, G. K., M.S. Thesis, Univ. of Texas, Austin, TX, 1967; also Himmelblau, D. M., *Applied Nonlinear Programming*, McGraw–Hill, New York, 1972, pp. 407–410.

[11]Gabriele, G. A., and Beltracchi, T. J., "OPT3.2: A FORTRAN Implementation of the Generalized Reduced Gradient Method," Users Manual, Dept. of Mechanical Engineering, Aerospace Engineering and Mechanics, Rensselaer Polytechnic Inst., Troy, NY, Jan. 1988.

[12]Balling, R. J., and Sobieszczanski-Sobieski, J., "Optimization of Coupled Systems: A Critical Overview of Approaches," *Proceedings of the AIAA/NASA/USAF/ISSMO 5th Symposium on Multidisciplinary Analysis and Optimization*, AIAA, Washington, DC, 1994, pp. 697–707 (AIAA Paper 94-4330).

[13]Sellar, R., Stelmack, M., Batill, S., and Renaud, J., "Response Surface Approximations for Discipline Coordination in Multidisciplinary Design Optimization," AIAA Paper 96-1383, April 1996.

[14]Sellar, R. S., "Multidisciplinary Design Using Artificial Neural Networks for Discipline Coordination and System Optimization," Ph.D. Dissertation, Aerospace and Mechanical Engineering Dept., Univ. of Notre Dame, Notre Dame, IN, April 1997.

[15]Sunar, M., and Belegundu, A. D., "Trust Region Methods for Structural Optimization Using Exact Second Order Sensitivity," *International Journal for Numerical Methods in Engineering*, Vol. 32, No. 2, 1991, pp. 275–293.

[16]Conn, A. R., Gould, N. I. M., and Toint, P. L., "Global Convergence of a Class of Trust Region Algorithms for Optimization with Simple Bounds," *SIAM Journal of Numerical Analysis*, Vol. 25, No. 2, 1988, pp. 433–464.

A. D. Belegundu
Associate Editor