Computational Study of a Nonhierarchical Decomposition Algorithm

Jayashree Shank, Calvin Ribbens, Raphael Haftka, and Layne Watson

TR 92-41

Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061

August 7, 1992

COMPUTATIONAL STUDY OF A NONHIERARCHICAL DECOMPOSITION ALGORITHM

Jayashree Shankar
Department of Computer Science

Calvin J. Ribbens
Department of Computer Science

Raphael T. Haftka
Department of Aerospace and Ocean Engineering

Layne T. Watson
Interdisciplinary Center for Applied Mathematics
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061-0106

Abstract

Optimizing the design of complex ground and flight vehicles involves multiple disciplines and multi-layered computer codes stitched together from mostly incompatible pieces. The application of established, large scale, optimization algorithms to the complete model is nearly impossible. Hierarchical decompositions are inappropriate for these types of problems and do not parallelize well. Sobieszczanski-Sobieski has proposed a nonhierarchical decomposition strategy for nonlinear constrained optimization that is naturally parallel. Despite some successes on engineering problems, the algorithm as originally proposed fails on simple two dimensional quadratic programs. This paper demonstrates the failure of the algorithm for quadratic programs, and suggests a number of possible modifications.

1. Introduction.

Many engineering problems involve large scale optimization over many different disciplines. For example, optimizing the design of complex terrestrial and aerospace vehicles involves multiple disciplines (structural mechanics, aerodynamics, thermodynamics, control theory, etc.) and disciplinary computer codes with thousands or millions of variables. Computer codes for the complete model are multi-layered and stitched together from various barely compatible pieces, making the application of established, large scale, optimization paradigms to the total model practically impossible. As is the case with many large scale problems, a decomposition of the problem into subproblems helps reduce the time and complexity of solution. The strategy governing the decomposition of a large scale problem can directly affect the ease and accuracy of the solution. The concept of a linear decomposition strategy [30] has been used with good results in a number of cases. This method works very well in the case of a system that is amenable to such a decomposition, i.e., when subsystems can be laid out clearly in a hierarchical fashion.

For a system with many interdependencies between the probable subproblems, using a linear decomposition strategy implies choosing one subsystem before another, thereby establishing an artificial hierarchy. The order chosen will affect the solution iterates, making this strategy ill suited or even nonconvergent for such nonhierarchic problems.

These considerations led Sobieski [29] to propose a new nonhierarchic decomposition strategy. Since nonlinear optimization can be reduced to a series of quadratic programs, it is appropriate to

study this new algorithm first on quadratic programs. Thus, this paper first studies the various tuning parameters occurring in this algorithm, using a model quadratic programming problem. A series of experiments shows that modifications to the algorithm as originally proposed by Sobieski [29] are necessary for convergence in general. This modified algorithm is then used to solve problems involving a number of subsystems, each with a varying number of design variables.

The tests are carried out on quadratic programming (QP) problems of different dimensions. The decomposition then yields subproblems which are also QP problems. The method employed to solve these smaller QP problems is elimination of variables [11]. Also optimization packages such as MINOS [22] and QPSOL [15] were used to verify the correct answers.

A detailed description of the original algorithm, modifications to it, tabulations of the results obtained for the test problems of different dimensions, and analysis of the results are presented.

2. Problem Statement.

Consider the following nonlinear programming problem (NLP),

$$\begin{aligned} &\min_{x} \Theta(x) \\ \text{subject to} \quad &g(x,y) \leq 0, \\ &h(x,y) = 0, \end{aligned}$$

where $x \in E^n$, $y \in E^p$, g is an m-dimensional vector function and h is a p-dimensional vector function. x is the set of design variables and y is the set of behavior variables which are the unknowns in each subsystem.

The approach (known as subspace optimization) is to solve this problem by solving a set of subproblems. To outline the differences between the current scheme and simple decomposition, we introduce the following terminology:

$$\begin{gathered} x = (X^1, X^2, \ldots, X^N), \qquad X^i \in E^{n_i}, \qquad n_1 + n_2 + \ldots + n_N = n, \\ y = (Y^1, Y^2, \ldots, Y^N), \qquad Y^i \in E^{p_i}, \qquad p_1 + p_2 + \ldots + p_N = p, \\ g = \begin{pmatrix} g^1 \\ \vdots \\ g^N \end{pmatrix}, \qquad h = \begin{pmatrix} h^1 \\ \vdots \\ h^N \end{pmatrix}, \\ h^i(x,y) \in E^{p_i}, \qquad g^i(x,y) \in E^{m_i}, \qquad m_1 + \ldots + m_N = m, \\ h^i(x,y) = Y^i - \tilde{h}^i\left(x, Y^1, \ldots, Y^{i-1}, Y^{i+1}, \ldots, Y^N\right). \end{gathered}$$

The sub vector X^i is the set of design variables corresponding to the ith subsystem. Similarly the sub vector Y^i is the set of behavior variables of the ith subsystem. For any vector function $f(x,y)$, let $\hat{f}(X^i, Y^i)$ denote f with all the components $X^1, \ldots, X^{i-1}, X^{i+1}, \ldots, X^N, Y^1, \ldots, Y^{i-1}, Y^{i+1}, \ldots, Y^N$ fixed. Note the assumption that each Y^i can be explicitly determined in terms of x and the other subvectors Y^j.

3. Simple Sequential Decomposition.

The approach is to first divide the given large problem into a set of independent subproblems, corresponding naturally to the subsystems comprising the larger system. The ith subsystem would be

$$\begin{aligned} &\min_{X^i} \hat{\Theta}(X^i) \\ \text{subject to} \quad &\hat{g}^i\left(X^i, Y^i\right) \leq 0, \\ &\hat{h}^i\left(X^i, Y^i\right) = 0, \end{aligned}$$

where the system of equalities $\hat{h}^i = 0$ is used to eliminate Y^i from $\hat{g}^i$. The subproblems are solved sequentially for $i = 1, \ldots, N$, with one pass through all the subsystems constituting one outer iteration. The outer iterations are repeated until the same point $(\bar{x}, \bar{y})$ solves all N subproblems. While solving the ith subsystem the values of $X^1, \ldots, X^{i-1}, X^{i+1}, \ldots, X^N, Y^1, \ldots, Y^{i-1}, Y^{i+1}, \ldots, Y^N$ are fixed. They can be chosen in a Gauss-Seidel manner where the first $i-1$ X and Y subvectors used have their latest values from solving the first $i-1$ subproblems. A parallel algorithm, solving the subproblems concurrently, would use a Jacobi scheme where the values of all the X^j and Y^j vectors are updated only at the end of each major outer iteration. The ensuing discussion assumes a Jacobi scheme.

4. Decomposition with Approximate Coupling.

In the scheme proposed by Sobieski [29], a measure of the constraints in each of the other subsystems is also brought into the ith subsystem in the form of one cumulative constraint C_i^k per subsystem. The approximate cumulative constraint C_i^k of the kth subsystem in the ith subsystem is obtained from the corresponding constraints $g^k \in E^{m_k}$ as a linearization of the Kreisselmeier-Steinhauser cumulative constraint

$$K_k(x, y) = \frac{1}{\rho} \ln \left(\sum_{j=1}^{m_k} e^{\rho g_j^k(x,y)} \right).$$

The ρ in the Kreisselmeier-Steinhauser function is a constant used to control the accuracy of the cumulative constraint approximation. The linearization of this cumulative constraint of the kth subsystem with respect to the variables of the ith subsystem is

$$C_i^k(X^i, Y^i) = \hat{K}_k(X_0^i, Y_0^i) + \sum_{j=1}^{n_i} \frac{\partial \hat{K}_k}{\partial X_j^i}(X_0^i, Y_0^i) \left(X_j^i - (X_0^i)_j \right).$$

In the ith subsystem the cumulative constraints of the other subsystems are brought in as constraints. Therefore, a violated cumulative constraint of one subsystem may be satisfied by decisions taken in every one of the other subsystems. Therefore, we introduce coefficients r_i^p to represent the fractional "responsibility" assigned to the ith subsystem for reducing the violation of the cumulative constraint of the pth subsystem, for each $p = 1, \ldots, N$. Thus we have N^2 r-coefficients. The r_i^p's are defined in such a way that

$$\sum_{i=1}^{N} r_i^p = 1,$$

Sobieski [29] suggested the initialization of the r-coefficients in such a way that they are proportional to the degree of influence exerted by the ith subsystem on the pth cumulative constraint. This initialization is discussed in the Appendix.

To further reduce the objective function we allow cumulative constraints to be violated in one subsystem, provided that the violation will be offset by oversatisfaction of that constraint in another subsystem. To account for such tradeoffs, we introduce the N^2 coefficients t_i^p, corresponding to

the cumulative constraint of the pth subsystem when present in the ith subsystem. For the pth cumulative constraint,

$$\sum_{i=1}^{N} t_i^p = 0,$$

maintains the constraint at a value of zero. This condition and the condition on the r-coefficients are enforced in what is called the coordination optimization phase, which is solved to update the values of the r's and the t's at the end of every outer iteration. The t_i^p's are initialized at the beginning of the algorithm to zero.

As has been described above, the r_i^p's are needed only in the case of a violation and the t_i^p's only when the constraints are critical, therefore only one of the two is needed at a time. Therefore we introduce N coefficients s^p which act as switches, one for each of the cumulative constraints of the subsystems. s^p is set to one (activating the r-coefficients) if the corresponding constraint $K_p \leq 0$ is violated at the outset of the system optimization procedure and stays at one until the K_p is driven to a critical status (zero value). Once K_p becomes critical, s^p is reset to zero (activating the t-coefficients) and stays at zero until the system optimization procedure terminates. The switch s^i is applied selectively to the natural constraints g^i of the ith subsystem (i.e., the constraints that are assigned to the ith subsystem) by multiplying the r-coefficient r_i^i by a factor of $\max\{\hat{g}^i(X_0^i, Y_0^i), 0\}$, so that constraints which are already satisfied are not taken into consideration.

Thus, the ith subsystem optimization problem is

$$\begin{aligned}
&\min_{X^i} \hat{\Theta}(X^i) \\
\text{subject to} \quad &\hat{g}^i(X^i, Y^i) \leq s^i \max\{\hat{g}^i(X_0^i, Y_0^i), 0\}(1 - r_i^i) + (1 - s^i)t_i^i, \\
&C_i^p(X^i, Y^i) \leq \hat{K}_p(X_0^i, Y_0^i)\, s^p(1 - r_i^p) + (1 - s^p)t_i^p, \\
&\qquad\qquad p = 1, \ldots, i-1, i+1, \ldots, N, \\
&\hat{h}^i(X^i, Y^i) = 0.
\end{aligned}$$

The constrained minimum of Θ obtained from each subsystem optimization is a function of the constants r_i^p and t_i^p, and its partial derivatives with respect to r_i^p and t_i^p (assuming they exist) can be computed from the expressions given in the Appendix using gradient information for the Θ and C functions. These derivatives are used for a linear approximation of Θ that is the objective function for the *coordination optimization phase*, the last (and synchronizing) step of an outer iteration.

The coordination optimization phase (COP) solves a linear program to adjust the coefficients r_i^p and t_i^p, so that the objective function Θ will be further reduced (if possible) at the end of the next outer iteration. The linear program uses a linear extrapolation of Θ based on the partial derivatives $\partial\Theta/\partial z$ described above. Here z represents either an r- or a t-coefficient. Move limits (upper and lower bounds U_i^p, $\tilde{U}_i^p$, L_i^p and $\tilde{L}_i^p$ for r_i^p and t_i^p, respectively) are needed to prevent large changes in the r- and t-coefficients caused by the nonlinearity of the original problem. For the first COP execution, the r_i^p's may be initialized as already suggested and the t_i^p's are initialized to zero. For every subsequent execution, the r_i^p's and the t_i^p's are initialized to the terminal values from the previous COP execution. The result of the COP execution is a new set of r_i^p's and t_i^p's to be used in the next outer loop of subsystem optimizations. The adjustment of the r_i^p's

and t_i^p's to the new values amounts to a reassignment of the responsibility for eliminating the constraint violations among the subsystems and to issuing a new set of instructions about trading the constraint violations/oversatisfactions among these subsystems. Let (x_0, y_0) be the current updated point (the result of the Jacobi outer iteration) and

$$\Theta_1 = \Theta(x_0, y_0) + \sum_{p=1}^{N}\sum_{i=1}^{N} \frac{\partial \Theta}{\partial r_i^p}\Delta r_i^p + \sum_{p=1}^{N}\sum_{i=1}^{N} \frac{\partial \Theta}{\partial t_i^p}\Delta t_i^p,$$

where $\Delta r_i^p = \left(r_i^p - (r_i^p)_0\right)$ and $\Delta t_i^p = \left(t_i^p - (t_i^p)_0\right)$. The partial derivatives $\partial\Theta/\partial r_i^p$ and $\partial\Theta/\partial t_i^p$ are evaluated at the optimal point computed by the ith subsystem optimization. Let

$$R = \left(r_1^1, r_1^2, \ldots, r_1^N, r_2^1, \ldots, r_2^N, \ldots, r_N^1, \ldots, r_N^N\right)$$

and

$$T = \left(t_1^1, t_1^2, \ldots, t_1^N, t_2^1, \ldots, t_2^N, \ldots, t_N^1, \ldots, t_N^N\right).$$

Then Θ_1 is a function of R and T. The linear program solved during the coordination optimization phase is:

$$\min_{R,T} \Theta_1(R,T)$$

$$\begin{aligned}
\text{subject to} \quad & \sum_{k=1}^{N} r_k^p = 1, && p = 1, \ldots, N, \\
& \sum_{k=1}^{N} t_k^p = 0, && p = 1, \ldots, N, \\
& 0 \le r_k^p \le 1, && p = 1, \ldots, N, \quad k = 1, \ldots, N, \\
& L_k^p \le r_k^p \le U_k^p, && p = 1, \ldots, N, \quad k = 1, \ldots, N, \\
& \tilde{L}_k^p \le t_k^p \le \tilde{U}_k^p, && p = 1, \ldots, N, \quad k = 1, \ldots, N.
\end{aligned}$$

5. Pseudocode for algorithm.

An algorithmic description of the whole process in pseudo-code is given next, using the following model quadratic programming problem (without the variables y and equality constraints $h(x, y) = 0$) for specificity:

$$\min_{x} \; x^t A x$$

$$\text{subject to} \quad Bx \le d,$$

where

$$A = \begin{pmatrix} A_{11} & \alpha_{12}A_{12} & \ldots & \alpha_{1N}A_{1N} \\ \alpha_{12}A_{12}^t & A_{22} & \ldots & \alpha_{2N}A_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{1N}A_{1N}^t & \alpha_{2N}A_{2N}^t & \ldots & A_{NN} \end{pmatrix}, \quad B = \begin{pmatrix} B_{11} & \beta_{12}B_{12} & \ldots & \beta_{1N}B_{1N} \\ \beta_{21}B_{21} & B_{22} & \ldots & \beta_{2N}B_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{N1}B_{N1} & \beta_{N2}B_{N2} & \ldots & B_{NN} \end{pmatrix},$$

$$d = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{pmatrix}, \qquad x = \begin{pmatrix} X^1 \\ X^2 \\ \vdots \\ X^N \end{pmatrix},$$

and $X^i \in E^{n_i}$, $A_{ij} \in E^{n_i \times n_j}$, $B_{ij} \in E^{p_i \times n_j}$, $d_i \in E^{p_i}$, the A_{ii} are symmetric and positive definite and the α_{ij}, β_{ij} are fixed "coupling" parameters, for all $i, j = 1, \ldots, N$. Using notation defined in the Appendix, pseudo-code for the algorithm applied to this quadratic programming problem (QP) is:

Choose an initial estimate x and initialize the r-, s- and t-coefficients;
Repeat until minimum reached
begin
  for $i = 1$ **to** N **do**
  begin
    Calculate the linearization $C_i^j(X^i)$ of the cumulative constraint for the jth subsystem, for all $j \neq i$;
    Calculate the ith subsystem's self responsibility

$$\delta^i = s^i \max\left\{\hat{g}^i\left(X_0^i, Y_0^i\right), 0\right\}\left(1 - r_i^i\right) + \left(1 - s^i\right) t_i^i;$$

    Solve the QP (ith subsystem)

$$\min_{X^i} \hat{\Theta}\left(X^i\right) = (X^i)^t A_{ii} X^i + 2\left(\sum_{j \neq i} \alpha_{ij} (X^i)^t A_{ij} X^j\right)$$

$$\text{subject to} \quad \sum_{j=1}^{N} \beta_{ij} B_{ij} X^j - d_i \leq \delta^i, \qquad (\beta_{ii} = 1)$$

$$\tilde{C}_i^j\left(X^i\right) \leq 0, \qquad \text{for all } j \neq i;$$

    Calculate (if not already available) the Lagrange multipliers λ using the method given in the Appendix.
    Calculate $\dfrac{\partial \Theta}{\partial r_i^j}$ and $\dfrac{\partial \Theta}{\partial t_i^j}$ for $j = 1, \ldots, N$;
  end
  Solve the LP (Coordination Optimization Phase)

$$\min_{R,T} \Theta_1(R, T)$$

$$\text{subject to} \quad \sum_{k=1}^{N} r_k^p = 1, \qquad \sum_{k=1}^{N} t_k^p = 0, \qquad p = 1, \ldots, N,$$

$$0 \leq r_k^p \leq 1, \qquad L_k^p \leq r_k^p \leq U_k^p, \qquad \tilde{L}_k^p \leq t_k^p \leq \tilde{U}_k^p,$$

$$p = 1, \ldots, N, \qquad k = 1, \ldots, N;$$

end (repeat)

6. Initial tests.

Testing of this algorithm was first performed on a simple 2×2 case:

Example 1.

$$\begin{aligned} \min_{x} \quad & x_1^2 + x_2^2 \\ \text{subject to} \quad & x_1 + \beta x_2 \leq 4, \\ & \beta x_1 + x_2 \geq 2, \\ \text{where } & x = (x_1, x_2)^t \in E^2. \end{aligned}$$

Here each constraint is taken to be in a subsystem by itself with $X^1 = (x_1)$ and $X^2 = (x_2)$. The results are tabulated in Table I. The column headings are the starting points, the last column gives the solutions for the different values of β, and each entry contains a convergence code and the number of iterations taken. The code IF means an infeasible subproblem is encountered at the very first iteration and the procedure is terminated, R means the solution is reached in the iteration indicated, but subsequently an infeasible subproblem is encountered, WR means a wrong point is reached before an infeasible subproblem causes termination, O means there is oscillation through the number of iterations indicated, WC means there is convergence to a point other than the solution, and NC means there is no convergence even after the number of iterations indicated.

TABLE I
Original algorithm applied to the 2×2 *case.*

β	(2,3)	(4,-1)	(1,-1)	(0.8,1.5)	(10,3)	solution
0.0	R 1	R 1	R 1	R 1	R 1	(0.0,2.0)
0.1	WR 1	IF 1	WC 4	WC 4	IF 1	(0.198,1.98)
0.3	WR 1	IF 1	WC 5	WC 4	IF 1	(0.55,1.835)
0.5	WR 1	IF 1	O 150	NC 150	WR 1	(0.8,1.6)
1.0	O 150	O 150	O 150	O 150	O 150	(1.0,1.0)

As can be seen from the table, the main problem with the algorithm is not being able to deal with infeasibility in a subproblem. This issue of infeasibility was mentioned in the appendix of [29], and is addressed in the next section. Another reason for the algorithm not being successful is the way the s-coefficient is set permanently to zero once a constraint becomes critical. When a constraint that was once critical becomes violated, the r-coefficients cannot be brought in to reduce the violation.

7. Modifications to the original algorithm.

Several modifications and variations of the original algorithm as described in [29] are discussed in detail in [26]. All these modifications were genuine attempts, based on rational considerations, to fix the failures of the algorithm. Probably the COP should be entirely redesigned, but that constitutes a completely new algorithm, and was not pursued here. These modifications, fully described and justified in [26], are listed next.

Natural Constraints.

The natural constraints of the subsystem being solved are used instead of the cumulative constraint representing them. The algorithm described in Section 4 includes this modification.

Changes in setting of the switch coefficients s^p.

Set the s^p coefficient at the end of every outer iteration depending on whether the corresponding constraint was satisfied or not.

Handling infeasibility.

Infeasibility in a subsystem is handled by introducing a new variable ω in each of the constraints and adding a large multiple of this variable to the objective function to be minimized.

Limit on t-coefficients.

Bounds on the t-coefficients t_k^p are variable and related to the sensitivity of the pth cumulative constraint to the variables of the kth subsystem.

Convergence criterion.

The convergence criterion involves the difference between three successive iteration values of the design vector and the difference between the values of the t- and r-coefficients.

Changes to the ρ coefficient.

After the required convergence criterion is met the ρ coefficient is increased and the whole process is repeated again to check if the convergence criterion is still met.

Cross derivatives.

The cross derivatives are checked to see if one subsystem is at all dependent on the variables of another subsystem, if not the corresponding r- and t-coefficients are fixed at zero. This is done at the end of every outer iteration.

Changes to the r-coefficients.

The diagonal r-coefficients r_i^i are assigned a minimum value of 0.2 always and a 20% move limit is imposed on all the r-coefficients.

No COP.

If the objective function of the COP is a constant then the COP is skipped for that iteration.

Resetting the t-coefficients.

After the COP a check is performed on the t-coefficients to see if for a particular p the sum of the contributions of all the corresponding coefficients to the objective function of the COP is zero. If so all the t coefficients corresponding to this p are forced to be zero.

8. Further tests.

The original algorithm as proposed by Sobieszczanski-Sobieski [29] did not prove to be successful as indicated by Table I. Results of two of the most successful variations to this algorithm tested on the 2×2 case are tabulated in Tables II and III. The characteristics of the algorithms used are given above the corresponding tables. "s updated" indicates that the s-coefficient is updated at the end of every outer iteration as indicated in the modifications given in Section 7. "ω used" means that an artificial variable ω was introduced to deal with infeasible subproblems. The inclusion or exclusion of the COP is in the case of an infeasibility in any subproblem. The limit on the magnitude of the t-coefficient is 1 initially and this bound is decreased using a factor of 0.8 as described in [26]. The most successful version was used for larger test problems like the 3×3 case with two subsystems and the 6×6 case with three subsystems. The tests were carried out for five different values of β and for five different starting points. The column headings are the starting

points, the last column gives the solutions for the different values of β, and each entry contains a convergence code, and the number of iterations until the two-norm of the change in (x, R, T) is less than 0.0001. For Tables II and III the number of iterations the limit on the t-coefficient is held fixed (at 1.0) before being reduced is 10, and 30 for Table IV. The code C means there is convergence to the solution, WC means there is convergence but not to the solution, and NC means there is no convergence even in the specified number of iterations.

TABLE II
s updated, ω used, no COP,
t bound at 1 and 0.8 update
after 10 iterations.

β	(2,3)	(4,-1)	(1,-1)	(0.8,1.5)	(10,3)	solution
0.0	C 6	C 6	C 6	C 6	C 6	(0.0,2.0)
0.1	C 71	WC 12	C 71	C 71	C 71	(0.198,1.98)
0.3	C 67	WC 25	C 64	C 67	C 67	(0.55,1.835)
0.5	C 66	WC 18	C 66	C 65	C 64	(0.8,1.6)
1.0	C 7	C 8	C 7	C 62	C 8	(1.0,1.0)

TABLE III
s updated, ω used, COP,
t bound at 1 and 0.8 update
after 10 iterations.

β	(2,3)	(4,-1)	(1,-1)	(0.8,1.5)	(10,3)	solution
0.0	C 6	C 6	C 6	C 6	C 6	(0.0,2.0)
0.1	C 71	C 67	C 71	C 71	C 71	(0.198,1.98)
0.3	C 67	C 72	C 64	C 67	C 67	(0.55,1.835)
0.5	C 66	C 65	C 66	C 65	C 64	(0.8,1.6)
1.0	C 7	C 8	C 7	C 62	C 8	(1.0,1.0)

Figures 1 and 2 are provided for a better understanding of the path taken from the starting point to the solution. The pictures correspond to Example 1 with $\beta = 0.1$. Figure 1 includes all the iterate values (except for the starting point) up to the solution. The first few segments are numbered $1 - 8$ at their midpoints. Figure 2 is a blown up view of the region of convergence that is marked in Figure 1. Some of the intermediate segments are numbered at their midpoints here. The solution is indicated by a $*$.

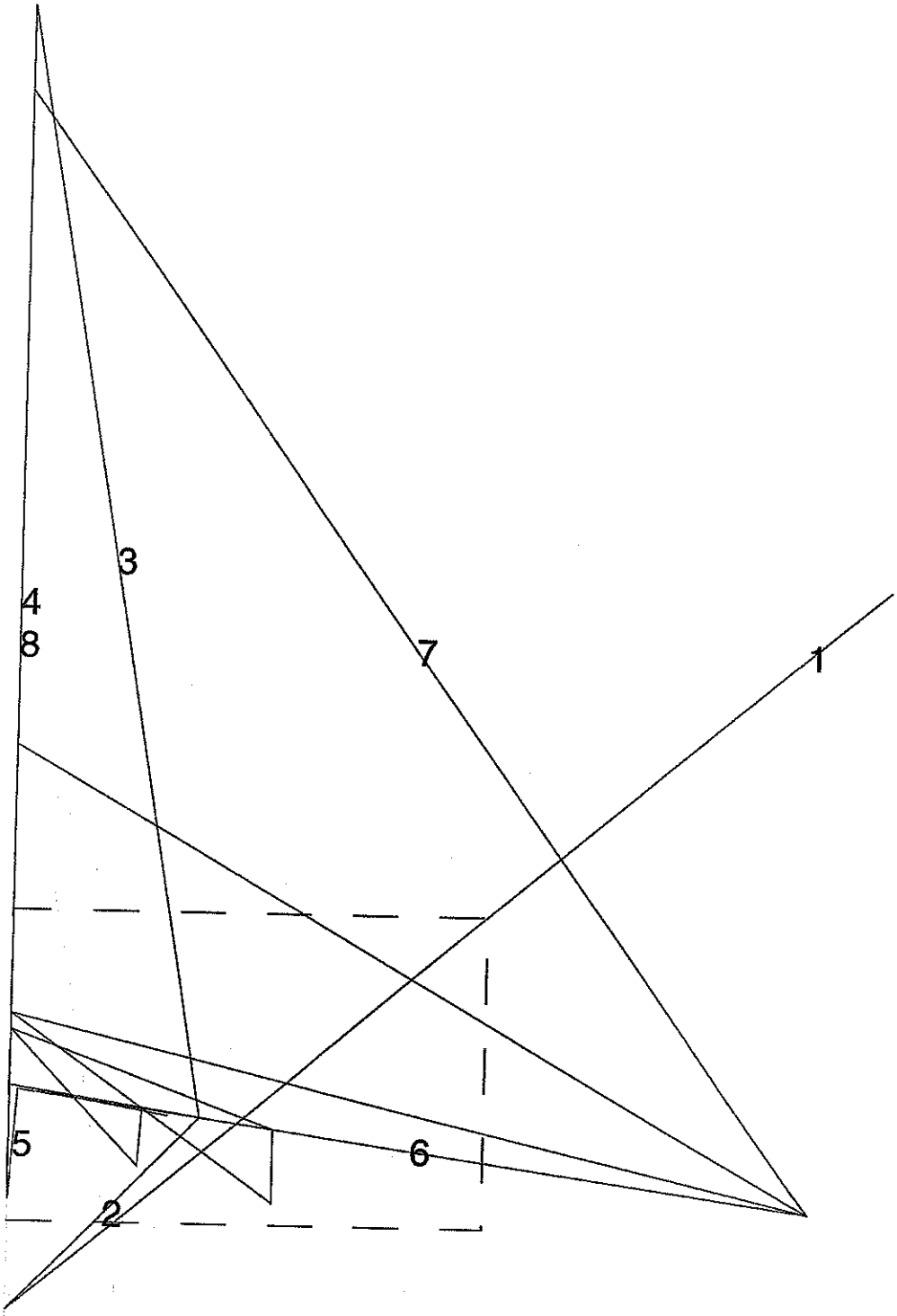

FIGURE 1. *Trace of the solution iterates for Example 1, corresponding to $\beta = 0.1$, with starting point (2,3) and solution (0.198,1.98).*

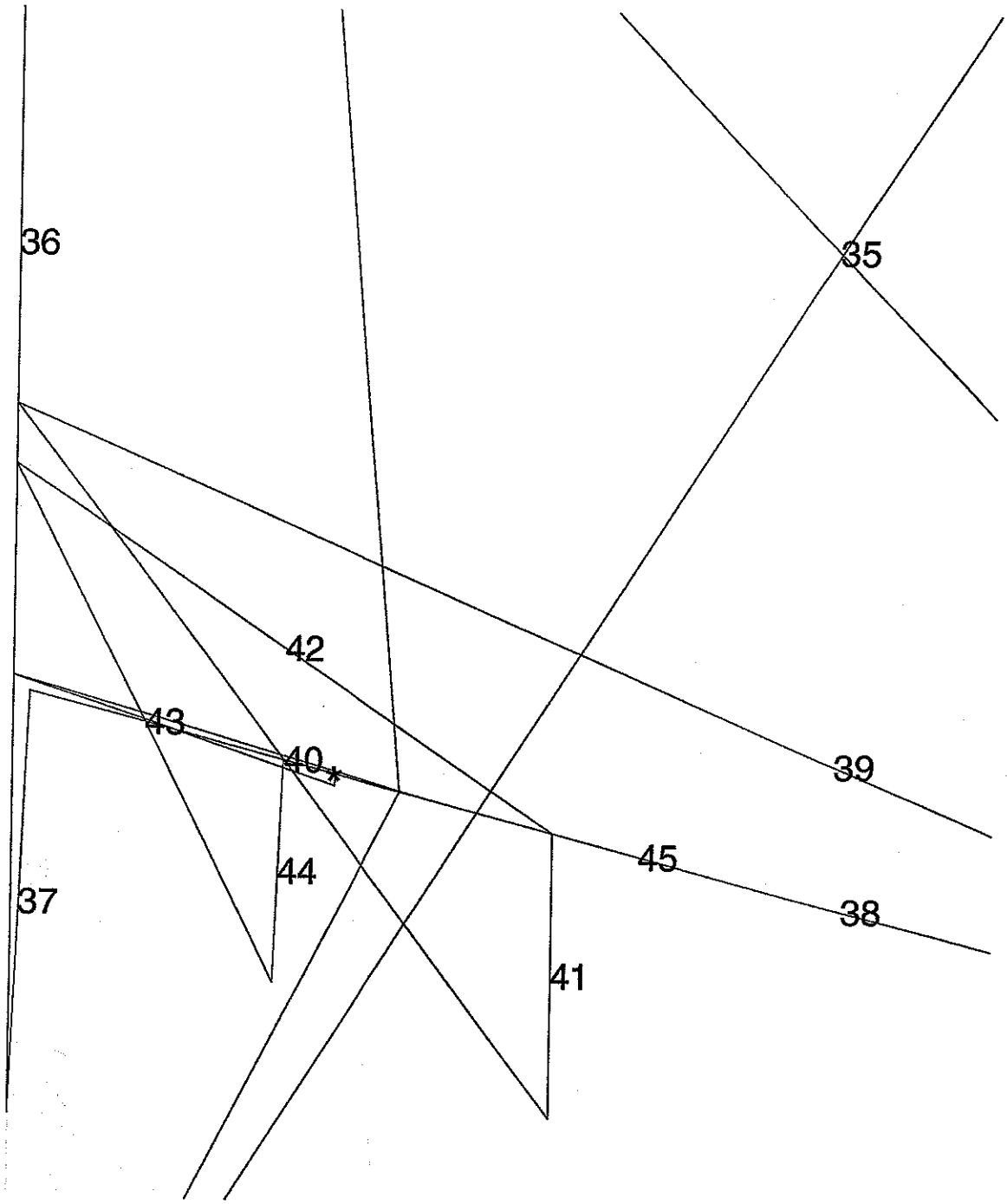

FIGURE 2. *Blown up view of the region of convergence of the plot shown in Figure 1.*

Example 2.

$$\begin{aligned} \min_x \quad & x_1^2 + x_2^2 + x_3^2 \\ \text{subject to} \quad & x_1 + x_2 + \beta x_3 \leq 4, \\ & -x_1 - x_2 - \beta x_3 \leq -2, \\ & -\beta x_1 - \beta x_2 - 5x_3 \leq -2, \\ & \text{where } x = (x_1, x_2, x_3)^t \in E^3 \end{aligned}$$

Here, the first two constraints belong to one subsystem and the third constraint to another subsystem, $X^1 = (x_1, x_2)$ and $X^2 = (x_3)$.

TABLE IV
s updated, ω used, COP,
t bound at 1 and 0.8 update
after 30 iterations.

β	(0,1,-3)		(1,1,0)		(4,0.1,0.8)		(-10,3,-10)		(0,0,0)		solution
0.0	C	6	C	6	C	6	C	6	C	6	(1,1,0.4)
0.1	C	86	C	89	C	98	C	89	C	87	(0.9819,0.9819,0.3607)
0.3	C	85	C	82	C	92	C	82	C	80	(0.9569,0.9569,0.2870)
0.5	C	93	C	81	C	81	C	89	WC	80	(0.8888,0.8888,0.4444)
1.0	WC	102	C	103	C	77	WC	102	C	76	(0.6666,0.6666,0.6666)

In Tables II–IV for $\beta = 0$, the 6 iterations corresponds to 3 iterations for the convergence test to be satisfied with 2 consecutive values of ρ; thus 6 actually is the smallest possible number of iterations, and corresponds to reaching the solution in one step. For the WC entries in Table IV, performing a cold start at the point they converge to results in convergence to the correct solution. Figure 3 corresponds to Example 2 and gives a trace of the iterates for $\beta = 0.5$ and starting point $(0, 1, -3)$ (not included in the picture).

Move limits on the x vector.

Introducing move limits on the x variables will help prevent oscillation of the iterates. The size of the move limits permitted can be varied depending on the problem. Let m be the move limit permitted and $\bar{x}_i$ the current value of x_i. Then the constraint

$$\bar{x}_i - m|\bar{x}_i| - 0.1 \leq x_i \leq \bar{x}_i + m|\bar{x}_i| + 0.1$$

is added for every component x_i of the x vector.

Tests were performed on the following 6×6 example using different move limits and the results are tabulated accordingly.

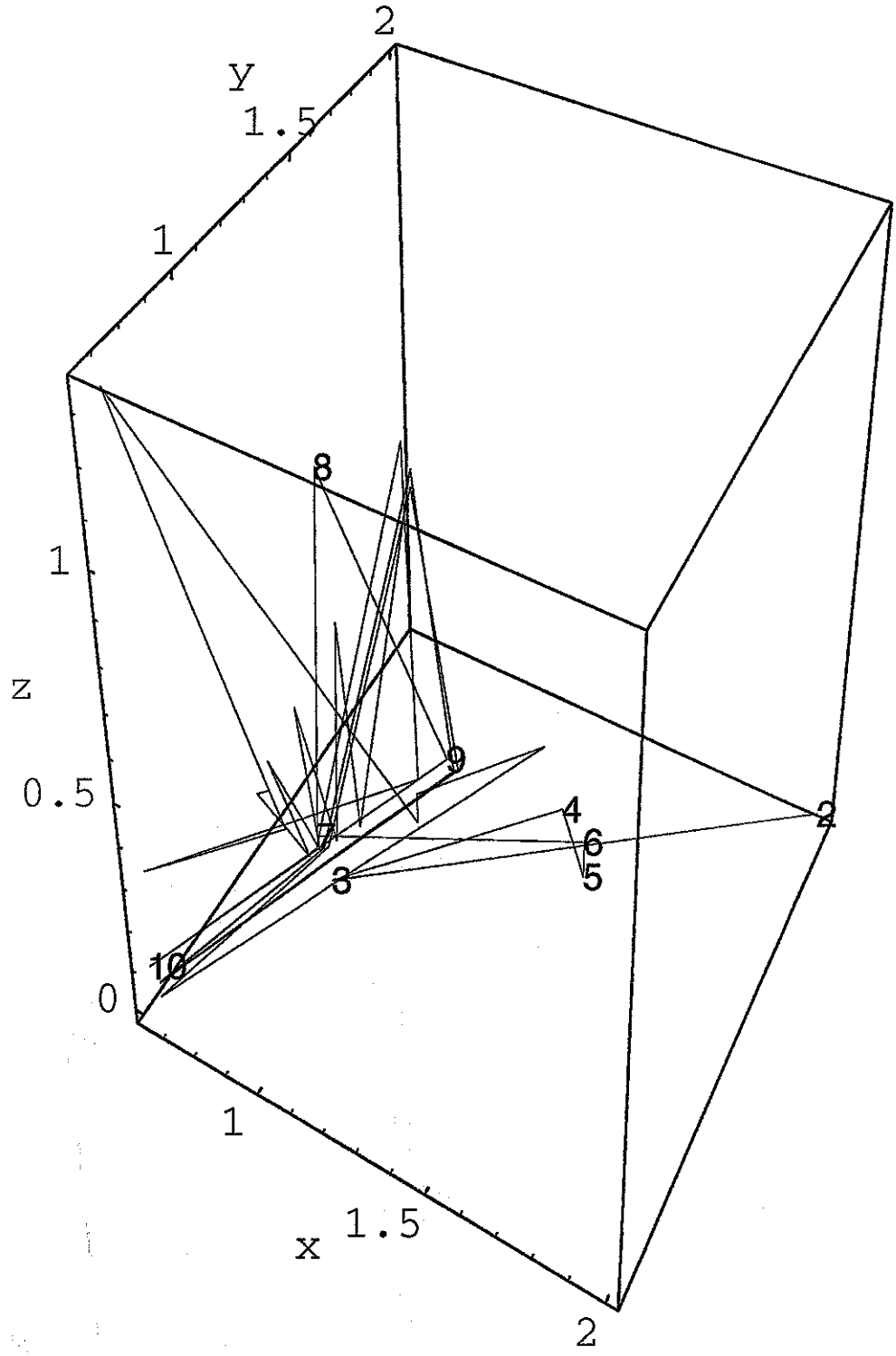

FIGURE 3. *Trace of the solution iterates for Example 2, corresponding to $\beta = 0.5$, with starting point (0,1,-3) and solution (0.88,0.88,0.44).*

Example 3.

$$\begin{aligned}
\min_x \quad & x_1^2 + x_2^2 + x_3^2 + 2.5x_4^2 + 2.5x_5^2 + 10x_6^2 \\
\text{subject to} \quad & x_1 + x_2 + x_3 + 0 - \beta x_5 - 2\beta x_6 \le 4, \\
& -x_1 - x_2 - x_3 - \beta x_4 + 0 + 0 \le -2, \\
& -x_1 - x_2 - 5x_3 + 0 + 0 + 0 \le -2, \\
& 0 + 0 + 0 + x_4 + x_5 - \beta x_6 \le -4, \\
& \beta x_1 + \beta x_2 + 0 - 5x_4 - 4x_5 - \beta x_6 \le 20, \\
& \beta x_1 + \beta x_2 - \beta x_3 + 0 + 0 - x_6 \le -6, \\
& \text{where } x = (x_1, x_2, x_3, x_4, x_5, x_6)^t \in E^6
\end{aligned}$$

Here there are three subsystems with $n_1 = 3$, $n_2 = 2$ and $n_3 = 1$.

If the algorithm as described thus far is used, the results are not encouraging as indicated by Table V. In Table V the number in each entry gives the number of iterations taken. The entries in this table correspond to a move limit of $m = 0.3$ on each component of the x vector. The limit on the t-coefficients is held fixed for 30 iterations before being reduced. There is convergence to the solution only in the case of $\beta = 0.0$; for larger βs the solution is not obtained for any of the starting points. An attempt to use the actual solution vector as the starting point also did not lead to convergence to the solution – the process actually diverged away from the solution. Clearly, this is a fundamental flaw.

The cause of the divergence was traced to the sensitivity information used in the COP. A thorough explanation, in the context of Example 3, of the failure of the COP to provide correct sensitivity information is offered in [26]. This suggests a complete redesign of the COP is called for, but the correct design is not apparent.

Table VI gives the results for the 6×6 problem if the sensitivities used in the COP are the average of the sensitivities obtained over two iterations. In each entry in Table VI, the number to the right of the convergence code indicates the number of iterations taken, the number below gives gives the number of iterations the limit on the t-coefficient is held fixed (at 1.0) before being reduced, and the third number gives the move limit permitted on each component of the x vector. While smoothed sensitivities clearly help (compare Tables V and VI), they are not a cure for the larger β cases. We remark that simply changing the move limits on x, r, and t is not the remedy, since hundreds of move limit variations were tried with results no better than those in Table VI.

TABLE V
s updated, ω used, COP,
t bounded at 1 and 0.8 update.

β	(0,0,0, 0,0,0)	(1,2,3, -1,1,5)	(-10,4,4, 0.8,0.1,1)	(1,1,1, 1,1,1)	(-4,2,2, 0,1,1)	solution
0.0	C 17	C 17	C 19	C 17	C 17	(0.6,0.6,0.6,-2.0,-2.0,6.0)
0.1	NC 154	NC 154	WC 113	NC 154	NC 151	(-2.4,-2.4,7.0,-1.7,-1.8,4.8)
0.3	NC 151	WC 153	WC 105	WC 105	NC 150	(-2.7,-2.7,8.0,-1.5,-1.8,1.9)
0.5	WC 104	WC 197	WC 104	WC 106	WC 106	(-1.7,-1.7,6.3,-1.5,-1.9,1.0)
1.0	WC 102	WC 105	WC 102	WC 102	WC 105	(-0.5,-0.5,4.2,-1.2,-2.0,0.7)

TABLE VI
s updated, ω used, COP,
t bounded at 1 and 0.8 update.

β	(0,0,0, 0,0,0)	(1,2,3, -1,1,5)	(-10,4,4, 0.8,0.1,1)	(1,1,1, 1,1,1)	(-4,2,2, 0,1,1)	solution
0.0	C 17 10 0.3	C 17 10 0.3	C 19 10 0.3	C 17 10 0.3	C 17 10 0.3	(0.6,0.6,0.6,-2.0,-2.0,6.0)
0.1	C 110 30 0.1	C 190 70 0.3	C 113 30 0.1	C 72 10 0.3	WC 109 30 0.1	(-2.4,-2.4,7.0,-1.7,-1.8,4.8)
0.3	WC 105 30 0.1	C 203 80 0.1	C 103 30 0.1	C 163 60 0.1	WC 103 30 0.1	(-2.7,-2.7,8.0,-1.5,-1.8,1.9)
0.5	WC 103 30 0.1	WC 103 30 0.1	WC 104 30 0.1	WC 107 30 0.1	WC 107 30 0.1	(-1.7,-1.7,6.3,-1.5,-1.9,1.0)
1.0	WC 103 30 0.1	WC 103 30 0.1	WC 104 30 0.1	WC 102 30 0.1	WC 102 30 0.1	(-0.5,-0.5,4.2,-1.2,-2.0,0.7)

9. Conclusions.

Despite the success reported by [8] with the original algorithm of Sobieski [29], we have required major modifications even for a 2×2 quadratic program. For larger QPs with weak subsystem coupling, the modified algorithm described in Section 7 works reasonably well. For larger QPs with strong coupling, the sensitivities obtained from the parallel subsystem optimizations were too unreliable for the COP to produce rational changes in the r- and t-coefficients. One possibility for correcting this problem may be to compute subsystem sensitivities with a global (and hence serial) step, as advocated in [25].

Several other observations are worth noting. Although some practitioners are willing to accept points with substantially improved objective function values that are not even local optima, the many instances of convergence to nonstationary points reported in the tables reinforce the belief that provable convergence properties are important.

The failure of the original algorithm, and numerous rational patches to it, on small QPs makes it extremely unlikely that the scheme would work on more general nonlinear programs (although with enough parameter tinkering we could get the optimal solution for *any given problem*). Admittedly, the coupling in the QPs is artificial, and the possibility remains that the coupling in practical multidisciplinary optimization problems (such as aeroelasticity) is such that the present decomposition algorithm works.

10. Acknowledgement.

This work was supported in part by NASA grant NAG-1-1079. The authors wish to thank the referees for some excellent suggestions.

Appendix.

Initialization of the r-coefficients.

The coefficients may be initialized on the basis of sensitivity information so as to assign a greater responsibility for a cumulative constraint satisfaction (of the ith subsystem say) to those subsystems that have a greater influence on that constraint. Let

$$(K_{pk})_i \equiv \frac{\partial K_p}{\partial X_i^k}(x_0, y_0).$$

Since $1 \le p \le N$, $1 \le k \le N$, and $1 \le i \le n_k$, there are Nn_k such partial derivatives $(K_{pk})_i$, for every k. Now define

$$a^{pk} \equiv \max_{1 \le i \le n_k} \left|(K_{pk})_i\right|, \qquad 1 \le p \le N, \quad 1 \le k \le N,$$

which measures the influence of the kth subsystem's variables X^k on the pth subsystem's constraints, as represented by K_p. Normalizing these N^2 influence coefficients gives the r-coefficients

$$r_k^p = \frac{a^{pk}}{\sum_{j=1}^{N} a^{pj}}, \qquad 1 \le p \le N, \quad 1 \le k \le N.$$

Optimum Sensitivity Analysis.

Let z denote either of r_i^p or t_i^p, and define the modified constraint functions

$$\tilde{g}^i(X^i, Y^i) = \hat{g}^i(X^i, Y^i) - \left[s^i \max\{\hat{g}^i(X_0^i, Y_0^i), 0\}(1 - r_i^i) + (1 - s^i)t_i^i\right],$$
$$\tilde{C}_i^p(X^i, Y^i) = C_i^p(X^i, Y^i) - \left[\hat{K}_p(X_0^i, Y_0^i)\, s^p(1 - r_i^p) + (1 - s^p)t_i^p\right],$$
$$i = 1, \ldots, N, \quad p = 1, \ldots, i-1, i+1, \ldots, N.$$

Let $\nabla_i = \left(\frac{\partial}{\partial X_1^i}, \ldots, \frac{\partial}{\partial X_{n_i}^i}\right)$,

$$G^i = \begin{pmatrix} \tilde{g}^i \\ \tilde{C}_i^1 \\ \vdots \\ \tilde{C}_i^{i-1} \\ \tilde{C}_i^{i+1} \\ \vdots \\ \tilde{C}_i^N \end{pmatrix},$$

and G_A^i denote the subvector of G^i corresponding to the active constraints at the current point. It is assumed that the dimension of G_A^i is less than or equal to n_i, and that the Jacobian matrix $\nabla_i G_A^i$ has full rank. Then the sensitivities of the minimum of Θ with respect to the constraints $\tilde{g}^i \le 0$, $\tilde{C}_i^p \le 0$ are given by the Lagrange multipliers

$$\lambda = -\left[(\nabla_i G_A^i)(\nabla_i G_A^i)^t\right]^{-1} (\nabla_i G_A^i)(\nabla_i \Theta)^t,$$

where everything is evaluated at the current point—the result of the Nth subsystem optimization. Now from this the sensitivities of the minimum of Θ with respect to the r_i^p and t_i^p are given by

$$\frac{\partial \Theta}{\partial z} = \lambda^t \frac{\partial G_A^i}{\partial z}.$$

Observe that from the form of G^i, the partials $\partial G_A^i / \partial z$ are trivial to compute. λ would not be computed explicitly from the projection operator as described above, but rather from a QR factorization of $\left(\nabla_i G_A^i\right)^t$, as described in Fletcher [11].

REFERENCES

[1] F. F. Abdi, H. Ide, V. J. Shankar, and J. Sobieszczanski-Sobieski, "Optimization for nonlinear aeroelastic tailoring criteria", in *Int. Coun. Aero. Sci.*, 16th Congress, Jerusalem, Aug., 1988.

[2] J.-F. M. Barthelemy and J. Sobieszczanski-Sobieski, "Optimum sensitivity derivatives of objective functions in nonlinear programming", *AIAA J.* vol. 21-6 pp. 913–915, 1983.

[3] J.-F. M. Barthelemy and M. F. Riley, "Improved multi-level optimization approach for the design of complex engineering systems", *AIAA J.* vol. 26-3 pp. 353–360, 1988.

[4] J.-F. M. Barthelemy, "Engineering applications of multilevel optimization", in *Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization*, Hampton, VA, Sept., 1988.

[5] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press: New York, 1982.

[6] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Prentice Hall: Englewood Cliffs, NJ, 1989.

[7] C. L. Bloebaum and P. Hajela, "Heuristic decomposition for non-hierarchic systems", in *AIAA/ASME/ASCE/AHS 32nd Structures, Structural Dynamics, and Materials Conference*, Baltimore, MD, April, 1991, pp. 344–353.

[8] C. L. Bloebaum, P. Hajela and J. Sobieszczanski-Sobieski, "Non-hierarchic system decomposition in structural optimization", in *Third NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization*, San Francisco, CA, Sept., 1990.

[9] A. C. Damota, "Analysis of parallelism for nonlinear programming", M.S. Thesis, Instituto de Pesquisas Espaciais, Sao Jose dos Campos.

[10] G. Di Pillo, F. Facchinei and L. Grippo, "A RQP algorithm using a differentiable exact penalty function for inequality constrained problems", Dept. of Information and Systems, Rome, Italy, Technical Report 03.89, 1989.

[11] R. Fletcher, *Practical Methods of Optimization, Vol. 2*, John Wiley and Sons: New York, 1980.

[12] J. C. Gilbert, "Maintaining the positive definiteness of the matrices in reduced secant methods for equality constrained optimization", IIASA, Laxenburg, Austria, IIASA working paper WP-87-123, 1987.

[13] P. E. Gill, W. Murray, M. A. Saunders and W. H. Wright, "Inertia-controlling methods for quadratic programming", *SIAM Review* vol. 33 pp. 1–36, 1991.

[14] P. E. Gill, W. Murray, M. A. Saunders and W. H. Wright, "Procedures for optimization problems with a mixture of bounds and general constraints", *ACM Transactions on Mathematical Software* vol. 10 pp. 282–298, 1984.

[15] P. E. Gill, W. Murray and M. A. Saunders, "QPSOL", Department of Operations Research, Stanford University, Stanford, CA, 1984.

[16] G. Kreisselmeier and R. Steinhauser, "Systematic control design by optimizing a vector performance index", in *Proc. IFAC Symp. on Computer Aided Design of Control Systems*, Zurich, Switzerland, 1979, pp. 113–117.

[17] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization", Dept. of EE and CS, Northwestern Univ., Evanston, IL, Technical Report NAM 03, 1988.

[18] M. Mckema and S. A. Zenios, "The optimal implementation of a quadratic transportation algorithm", in *Proc. 4th SIAM Conference on Parallel Processing for Scientific Computing*, Chicago, IL, Dec., 1989.

[19] B. Mendelson and I. Koren, "Estimating the potential parallelism and pipelining of algorithms for data flow machines", *J. Parallel Distributed Comput.* vol. 14 pp. 15–28, Jan. 1992.

[20] B. A. Murtagh and M. A. Saunders, "A projected Lagrangian algorithm and its implementation for sparse nonlinear constraints", *Math. Prog. Study* vol. 16 pp. 84–117, 1982.

[21] B. A. Murtagh and M. A. Saunders, "Large scale linearly constrained optimization", *Math. Prog.* vol. 14 pp. 41–72, 1978.

[22] B. A. Murtagh and M. A. Saunders, "MINOS 5.0 User's Guide", Dept. of Operations Research, Stanford Univ., Stanford, CA, Report SOL 83-20, 1983.

[23] E. R. Panier and A. L. Tits, "On feasibility, descent and superlinear convergence in inequality constrained optimization", Systems Research Center, Univ. of Maryland, College Park, MD, Technical Research Report 89-27, 1989.

[24] A. R. Parkinson, R. J. Balling, A. Wu and J. C. Free, "A general strategy for decomposing large design problems based on optimization and statistical test plans", in *Int. Conf. on Engineering Design, ICED 87*, Boston, MA, Aug., 1987, p. 162.

[25] J. E. Renaud and G. A. Gabriele, "Sequential global approximation in non-hierarchic system decomposition and optimization", *Advances in Design Automation* vol. 1 pp. 191–200, 1991.

[26] J. Shankar, "Analysis of a nonhierarchical decomposition algorithm", M.S. Thesis, Dept. of Computer Science, VPI&SU, Blacksburg, VA, August, 1992.

[27] Y. S. Shin, R. T. Haftka, L. T. Watson, and R. H. Plaut, "Tracing structural optima as a function of available resources by a homotopy method", *Comput. Methods Appl. Mech. Engrg.* vol. 70 pp. 151–164, 1988.

[28] Y. S. Shin, R. T. Haftka, L. T. Watson, and R. H. Plaut, "Design of laminated plates for maximum buckling load", *J. Composite Materials* vol. 23 pp. 348–369, 1989.

[29] J. Sobieszczanski-Sobieski, "Optimization by decomposition: a step from hierarchic to nonhierarchic systems", in *Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization*, Hampton, VA, Sept., 1988.

[30] J. Sobieszczanski-Sobieski, "A linear decomposition method for large optimization problems", NASA TM 83248, Feb., 1982.

[31] J. Sobieszczanski-Sobieski, B. B. James, and M. F. Riley, "Structural sizing by generalized, multilevel optimization", *AIAA J.* vol. 25-1 pp. 139–145, 1987.

[32] J. Sobieszczanski-Sobieski, "On the sensitivity of complex, internally coupled systems", in *AIAA/ASME/ASCE/AHS 29th Structures, Structural Dynamics, and Materials Conference*, Williamsburg, VA, April, 1988.

[33] J. Sobieszczanski-Sobieski, J.-M. F. Barthelemy, and K. M. Riley, "Sensitivity of optimum solutions to problem parameters", *AIAA J.* vol. 20 pp. 1291–1299, 1982.

[34] J. Sobieszczanski-Sobieski, "Two alternative ways for solving the coordination problem in multilevel optimization", NTIS HC/MF A03.

[35] O. O. Storaasli, and E. A. Carmora, "Parallel methods on large scale structural analysis and physics applications", *Computing Systems in Engineering* vols. 2-2 & 2-3 pp. 199, 1991.

[36] P. Tseng, "On the rate of convergence of a partially asynchronous gradient projection algorithm", *SIAM J. Optim.* vol. 1 pp. 603–619, 1991.

[37] G. N. Vanderplaats, Y. J. Yang, and D. S. Kim, "An efficient multilevel optimization method for engineering design", in *AIAA/ASME/ASCE/AHS 29th Structures, Structural Dynamics, and Materials Conference*, Williamsburg, VA, April, 1988, pp. 125–132.

[38] G. N. Vanderplaats, and H. Miura, "Computational trends in large scale engineering optimization", in *Applications of supercomputers in engineering: Fluid flow and stress analysis applications; Proceedings of the first international conference*, Elsevier/Computational Mechanics Publications, 2, Southampton, England, Sept., 1989, pp. 269–283.

[39] G. N. Vanderplaats, and H. Miura, "Experiences in large scale structural design optimization", in *Applications of supercomputers in engineering: Fluid flow and stress analysis applications; Proceedings of the first international conference*, Elsevier/Computational Mechanics Publications, 2, Southampton, England, Sept., 1989, pp. 251–268.

[40] G. Vasudevan, L. T. Watson, and F. H. Lutze, "A homotopy approach for solving constrained optimization problems", Dept. of Computer Sci., VPI&SU, Blacksburg, VA, Tech. Rep. 88–50, 1988.

[41] L. T. Watson, R. T. Haftka, F. H. Lutze, R. H. Plaut, and P. Y. Shin, "The application of globally convergent homotopy methods to nonlinear optimization", in *Advances in Numerical Partial Differential Equations and Optimization*, S. Gómez, J. P. Hennart, and R. A. Tapia (eds.), SIAM, Philadelphia, PA, 1991, pp. 284–298.

[42] L. T. Watson, S. C. Billups, and A. P. Morgan, "HOMPACK: A suite of codes for globally convergent homotopy algorithms", Dept. of Industrial and Operations Eng., Univ. of Michigan, Ann Arbor, MI, Tech. Rep. 85–34, 1985, and *ACM Trans. Math. Software* vol. 13 pp. 281–310, 1987.

[43] L. T. Watson and W. H. Yang, "Optimal design by a homotopy method", *Applicable Anal.* vol. 10 pp. 275–284, 1980.

[44] G. A. Wrenn and A. R. Dovi, "Multilevel decomposition approach to the preliminary sizing of a transport aircraft wing", in *AIAA/ASME/ASCE/AHS 29th Structures, Structural Dynamics, and Materials Conference*, Williamsburg, VA, April, 1988.