Convergence of trust region augmented Lagrangian methods using variable fidelity approximation data

José F. Rodríguez   John E. Renaud
Department of Aerospace and Mechanical Engineering
University of Notre Dame
Notre Dame, Indiana 46556

Layne T. Watson
Departments of Computer Science and Mathematics
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061-0106

Abstract

To date the primary focus of most constrained approximate optimization strategies is that application of the method should lead to improved designs. Few researchers have focused on the development of constrained approximate optimization strategies that are assured of converging to a Karush-Kuhn-Tucker (KKT) point for the problem. Recent work by the authors based on a trust region model management strategy has shown promise in managing the convergence of constrained approximate optimization in application to a suite of single level optimization test problems. Using a trust-region model management strategy, coupled with an augmented Lagrangian approach for constrained approximate optimization, the authors have shown in application studies that the approximate optimization process converges to a KKT point for the problem. The approximate optimization strategy sequentially builds a cumulative response surface approximation of the augmented Lagrangian which is then optimized subject to a trust region constraint. In this research the authors develop a formal proof of convergence for the response surface approximation based optimization algorithm. Previous application studies were conducted on single level optimization problems for which response surface approximations were developed using conventional statistical response sampling techniques such as central composite design to query a high fidelity model over the design space. In this research the authors extend the scope of application studies to include the class of multidisciplinary design optimization (MDO) test problems. More importantly the authors show that response surface approximations constructed from variable fidelity data generated during concurrent subspace optimizations (CSSOs) can be effectively managed by the trust region model management strategy. Results for two multidisciplinary test problems are presented in which convergence to a KKT point is observed. The formal proof of convergence and the successful MDO application of the algorithm using variable fidelity data generated by CSSO are original contributions to the growing body of research in MDO.

1 Nomenclature

k	Lagrangian iteration
s	approximate minimization iteration
i, j, l	variable indices
m	number of inequality constraints
n	number of design variables
p	number of equality constraints
$f(\mathbf{x})$	objective function
$\mathbf{g}(\mathbf{x})$	inequality constraint vector
$g_j(\mathbf{x})$	jth inequality constraint
$\mathbf{h}(\mathbf{x})$	equality constraint vector
$h_i(\mathbf{x})$	ith equality constraint
$\mathbf{c}(\mathbf{x})$	generalized constraint vector
$c_i(\mathbf{x})$	ith generalized constraint
c_1, c_2, c_3, c_4	real constants
$m(\mathbf{x})$	approximate model
$q(\mathbf{x})$	piecewise approximation
r_p	penalty parameter
t, t_1, t_2	step size length
$\mathbf{x}$	design vector, dimension n
x_l	lth design variable
$\mathbf{x}^U$	upper bound vector, dimension n
x_l^U	lth design upper bound
$\mathbf{x}^L$	lower bound vector, dimension n
x_l^L	lth design lower bound
$\mathbf{B}$	approximation of the Hessian
K	constraints residual
S	design space
$\beta, \beta_1, \beta_2, \kappa$	scalars
ϵ_1, ϵ_2	convergence tolerances
$\gamma_0, \gamma_1, \gamma_2, \eta, \mu$	trust region parameters
$\boldsymbol{\lambda}$	Lagrange multiplier vector, dimension m+p
λ_i	ith Lagrange multiplier
ρ	trust region ratio
$\Psi(\mathbf{x})$	alternative form for ine-

	quality constraints
$\Phi(\mathbf{x},\boldsymbol{\lambda}, r_p)$	augmented Lagrangian function
$\widetilde{\Phi}(\mathbf{x},\boldsymbol{\lambda}, r_p)$	approximation of the augmented Lagrangian function
Υ	fidelity control
$\Vert\cdot\Vert$	Euclidean norm
$\langle\,,\,\rangle$	inner product
∇	gradient operator with respect to design vector $\mathbf{x}$
$P(\mathbf{y}(\mathbf{x}))$	Projection operator. Projects the vector $\mathbf{y}$ onto the set of feasible directions at $\mathbf{x}$
Δ	trust region radius
$\Delta\mathbf{x}$	step size

2 Introduction

A common engineering practice is the use of approximation models in place of expensive computer simulations to drive a multidisciplinary design process based on nonlinear programming techniques. There has been a growing interest in the use of response surface approximations coupled with numerical optimizers to optimize large multidisciplinary systems. While high fidelity computational models exist in many disciplines, the high cost of computation associated with invoking these models prevents them from being used in optimization procedures where reaching a solution may require hundreds of analysis calls. The growing use of response surface approximation strategies is designed to reduce the number of detailed, costly computer simulations required during design optimization. In Renaud and Gabriele (1991, 1993, 1994), and Wujek *et al.* (1996, 1997) response surface approximations of multidisciplinary systems are developed using design data generated during concurrent subspace optimizations (CSSOs). In Swift and Batill (1991, 1992a,b, 1993) and Sellar *et al.* (1994, 1996a,b,c) response surface approximations are constructed using artificial neural networks for system approximation. In Giunta *et al.* (1994), Balabanov *et al.* (1996), Venter *et al.* (1996), and Roux *et al.* (1996) traditional response surface approximations are constructed for use in approximating aerodynamic and structural quantities. In Chen *et al.* (1995, 1996a,b), Lautenschlager *et al.* (1996), Koch *et al.* (1996), and Simpson *et al.* (1996), the use of response surface approximations for robust concept exploration and concurrent design are investigated. While these studies explore a diverse set of multidisciplinary design issues, they share the common attribute of using response surface approximations in place of expensive computer simulations to drive a design process based on nonlinear programming techniques.

In many of these studies a model management or move limit strategy is employed in an effort to insure that design decisions made based on lower fidelity information (i.e., response surface approximations) will yield improvements in the actual system. For example in Wujek *et al.* (1997) fixed percentage move limits are used to manage the approximate design optimization. Examples of other heuristic adaptive move limit strategies that have been developed in the literature include Fadel *et al.* (1990), Thomas *et al.* (1992), Chen (1993), Bloebaum *et al.* (1994), Grignon and Fadel (1994), and Pourazady and Fu (1996). In most cases these model management or move limit strategies are heuristic and in general lead to improved designs but not necessarily converged Karush-Kuhn-Tucker designs.

In fact in each of the aforementioned studies the primary consideration has been that the application of the approximate optimization strategies should lead to improved designs. Improved designs are obtained in each of the approximate optimization studies as measured by the design objective employed for each respective study. The attribute of achieving improved designs is laudable and obviously relevant for practicing designers. In this research an approximate optimization method is developed that not only leads to improved designs, but one that is shown to converge to a KKT point for the problem in implementation studies. The algorithm extends the work of several researchers from the mathematical community (Dennis and Torczon 1996; Alexandrov 1996; and Lewis 1996) who have reported success in using trust region methods for managing the convergence of *unconstrained* approximate optimization problems.

In this research a trust region approach for lower fidelity approximate model management is extended to the class of *constrained* approximate optimization problems to manage convergence. The extension developed in this research is similar to the strategy developed in Conn *et al.* (1988a,b, 1991), where trust regions are used to manage the convergence of a high fidelity augmented Lagrangian optimization. In this research a similar trust region approach is used to manage the convergence of low fidelity response surface approximations in an augmented Lagrangian optimization. A cumulative response surface approximation of the augmented Lagrangian is sequentially optimized subject to a trust region constraint which insures convergence of the strategy.

Note that the approximation strategy implemented in the augmented Lagrangian approach used in this research is fundamentally different than that of Conn *et al.* (1988). In Conn *et al.* it is recommended that exact Hessian information of the augmented Lagrangian be used during the optimization procedure. This is obviously prohibitive in application to large multidisciplinary design problems. In cases where exact Hessian information is not available, the Conn *et al.* algorithm builds an approximation of the Hessian terms using a BFGS approach. In Conn *et al.* a proof of convergence is developed for the BFGS based augmented Lagrangian approach. The Conn proof of convergence makes use of a trust region model management strategy that limits design moves to insure accuracy of the BFGS second order approximation.

In this research a cumulative response surface approximation (Rodríguez *et al.*, 1997) of the augmented Lagrangian is developed using second order response surface approximations of the objective functions and each of the constraints. This cumulative response surface approximation *replaces* the BFGS based quadratic approximation used in the Conn *et al.* (1988) approach. The decision to use response surface approximations in place of the Conn *et al.* (1988) BFGS approximation is philosophical and based on the objective of developing a proof of convergence for response surface based approximate optimization. In this research a trust region model management strategy is employed to limit design moves to regions that are accurately mapped by the cumulative response surface approximation.

Preliminary results of this research were just reported in Rodríguez *et al.*, 1997. The primary contribution of the preliminary study was to confirm that a trust region model management strategy is effective in managing the convergence of the sequential approximate optimization of the augmented Lagrangian where cumulative response surface approximations are used.

In this research a formal proof of convergence is developed for the response surface based constrained approximate optimization algorithm. A formal proof is significant in that it provides a mathematical foundation upon which approximate optimization strategies can continue to be developed. We are not claiming that a proof of convergence will have a significant impact on the practicing designer who is under pressure to produce improved designs in reduced design cycle times. We are instead focused on the fundamental validation that a proof of convergence provides in the development of approximate optimization algorithms.

The Rodríguez *et al.*, 1997 application studies were conducted on a suite of single level optimization problems. In this research the scope of application studies is extended to include the class of multidisciplinary design optimization (MDO) test problems. In the earlier studies of Rodríguez *et al.*, 1997 a high fidelity model over the design space was queried directly using a central composite design sampling strategy to build a response surface approximation over the design space during each iteration of the algorithm. Querying the high fidelity model many times during each iteration can itself become cost prohibitive as the size of the design space grows. This is particularly true for multidisciplinary systems in which iterative solution strategies are required to converge multiple disciplines. In this research the high fidelity model is no longer queried directly to generate response surface approximations, similar in spirit to the way Burgee *et al.* (1996), and Giunta *et al.* (1997), construct response surface approximations to low and medium fidelity data. In this research we have elected to use the design sampling strategy detailed in the CSSO approach of Wujek *et al.* (1996). Instead of querying the high fidelity model directly using a statistical sampling strategy, the Wujek approach queries approximation models available at the discipline design level. These models are of variable fidelity and will be discussed in detail in a later section. The sampling strategy in the Wujek *et al.* (1996) approach is optimization based. The design sites visited during the subspace optimizations are the response sampling sites. The resulting response surface is therefore constructed from data of variable fidelity. The results presented in this research show that response surface approximations constructed from variable fidelity data generated during concurrent subspace optimizations (CSSOs) can be effectively managed by the trust region model management strategy.

3 Augmented Lagrangian background

The following problem is considered:

$$\min\ f(\mathbf{x}) \tag{1}$$

subject to

$$h_i(\mathbf{x}) = 0, \quad i = 1 \ldots p, \tag{2}$$

$$g_j(\mathbf{x}) \geq 0, \quad j = 1 \ldots m, \tag{3}$$

$$x_l^L \leq x_l \leq x_l^U, \quad l = 1 \ldots n. \tag{4}$$

Here f, h_i, and g_j map R^n into R and the region $S = \{\mathbf{x} \,|\, \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U\}$ defined by (4) is assumed to have a nonempty interior and be compact. It will also be assumed that the functions f, h_i, and, g_j are twice continuously differentiable on an open set containing S.

An important class of minimization algorithms used to solve this problem is known as the *augmented Lagrangian method*. This method was proposed independently by Hestenes (1969), and Powell (1969), for the case in which only equality constraints are present. An extension of this method where inequality constraints are present was made by Rockafellar (1973). For the general case, the augmented Lagrangian Φ is given by

$$\Phi(\mathbf{x}, \boldsymbol{\lambda}, r_p) = f(\mathbf{x}) + \sum_{i=1}^{p} \big(\lambda_i h_i(\mathbf{x}) + r_p h_i(\mathbf{x})^2\big) + \sum_{j=1}^{m} \big(\lambda_{p+j} \Psi_j(\mathbf{x}) + r_p \Psi_j(\mathbf{x})^2\big), \tag{5}$$

where the λ_is are the Lagrange multipliers, r_p is the penalty parameter, and the function $\Psi(\mathbf{x})$ is defined as

$$\Psi_j(\mathbf{x}) = \min\left\{ g_j(\mathbf{x}), -\frac{\lambda_{p+j}}{2r_p} \right\}. \tag{6}$$

This formulation, suggested by Rockafellar (1973), avoids the use of slack variables and thus the dimension of the problem remains unchanged. One of the disadvantages of such a formulation is the discontinuity in the second derivative which limits the use of second order methods in the solution of the unconstrained problem. Details of the continuity of the first derivative of the function $\Psi(\mathbf{x})$ are given in Rodríguez *et al.* (1997).

When an optimization problem is solved using the method of multipliers (i.e., *augmented Lagrangian*) two important ingredients are needed. The first is updating of the Lagrange multipliers and the second is updating of the penalty parameter. Algorithms based on the *augmented Lagrangian* differ in the way these parameters are updated.

A simple Lagrange multiplier update formula, suggested by Hestenes and Powell for the case in which only the equality constraints are present, is given as

$$\lambda_i^{k+1} = \lambda_i^k + 2 r_p h_i(\mathbf{x}^k). \tag{7}$$

For the case in which inequality constraints are also present, their multipliers can be updated using a similar formula

$$\lambda_{p+j}^{k+1} = \lambda_{p+j}^k + 2 r_p \min\left\{ g_j(\mathbf{x}^k), -\frac{\lambda_{p+j}^k}{2r_p} \right\}. \tag{8}$$

The above update formulas require only function values rather than first order information, thus avoiding the solution of a system of equations to determine the multipliers.

The role of the penalty parameter in the scheme is to speed up convergence when the infeasibility of the constraints (2) and (3) converges to zero at a very low rate. In general, we can have one penalty parameter for each constraint, but many authors use a single parameter. In the method described in this research, a single penalty parameter is used, however, generalization to $m + p$ parameters is straightforward.

4 Convergence criteria

In this section we state some concepts which will be used later. Consider the problem

$$\min_{\mathbf{x}^L \le \mathbf{x} \le \mathbf{x}^U} F(\mathbf{x}) \tag{9}$$

It is known that problem (9) has a unique solution $\mathbf{x}^*$ whenever $F(\mathbf{x})$ has a minimum and is strictly convex in the region $S = \{\mathbf{x} \,|\, \mathbf{x}^L \le \mathbf{x} \le \mathbf{x}^U\}$, $\mathbf{x}^L < \mathbf{x}^U$. If we allow the components of the vectors $\mathbf{x}^L$ and $\mathbf{x}^U$ to be infinite, then this result holds for unbounded S.

Any solution to (9) satisfies the Karush-Kuhn-Tucker conditions, which imply:

$$\begin{array}{lll} (\nabla F(\mathbf{x}))_l = 0 & \text{if} & x_l^L < x_l < x_l^U, \\ (\nabla F(\mathbf{x}))_l \ge 0 & \text{if} & x_l = x_l^L, \\ (\nabla F(\mathbf{x}))_l \le 0 & \text{if} & x_l = x_l^U. \end{array} \tag{10}$$

Based on this, the solution to problem (9) can be characterized in terms of the projected gradient $P(\nabla F(\mathbf{x}))$, defined as:

$$P\big(\nabla F(\mathbf{x})\big)_l = \begin{cases} (\nabla F(\mathbf{x}))_l & \text{if} \quad x_l^L < x_l < x_l^U, \\ \min\big\{(\nabla F(\mathbf{x}))_l, 0\big\} & \text{if} \quad x_l = x_l^L, \\ \max\big\{(\nabla F(\mathbf{x}))_l, 0\big\} & \text{if} \quad x_l = x_l^U. \end{cases} \tag{11}$$

The conditions given in (10) and the definition (11) of the projected gradient show that $P(\nabla F(\mathbf{x}^*)) = 0$ whenever $\mathbf{x}^*$ is a solution of problem (9).

Convergence to a Karush-Kuhn-Tucker point in the augmented Lagrangian approach results from requiring that the residual of the constraints K be reduced after every iteration of the unconstrained minimization algorithm, where the residual of the constraints is defined as:

$$K = \max\Big\{ \max_{1\le i\le p} |h_i| \,,\, \max_{1\le j\le m} |\Psi_j| \Big\}. \tag{12}$$

The iterative process of augmented Lagrangian minimization is stopped when the following criteria are satisfied:

$$\|P(\nabla\Phi(\mathbf{x}^k, \boldsymbol{\lambda}^k, r_p^k))\| \le \epsilon_1 \quad \text{and} \quad K \le \epsilon_2, \tag{13}$$

where $\mathbf{x}^k$, $\boldsymbol{\lambda}^k$ denotes the k^{th} iterate obtained by the algorithm. This assumes that there exists $\mathbf{x}^k \in S$, $\boldsymbol{\lambda}^k$, and $r_p^k > 0$ such that (13) holds.

5 Trust region model management

Convergence to a minimum point in this research is achieved by successively building and then optimizing response surface approximations of the augmented Lagrangian. This creates the necessity of having a procedure to quantify the quality of these response surface approximations. One would like to be able to define the region in which the response surface approximation is reliable. To do this we will use a trust region approach.

A trust region test can be used to evaluate the "validity" of the current augmented Lagrangian response surface approximation $\widetilde{\Phi}(\mathbf{x}, \boldsymbol{\lambda}, r_p)$. This test is based on the trust region ratio ρ^s defined by

$$\rho^s = \frac{\Phi(\mathbf{x}^k, \boldsymbol{\lambda}^k, r_p^k) - \Phi(\mathbf{x}^{k,s}, \boldsymbol{\lambda}^k, r_p^k)}{\widetilde{\Phi}(\mathbf{x}^k, \boldsymbol{\lambda}^k, r_p^k) - \widetilde{\Phi}(\mathbf{x}^{k,s}, \boldsymbol{\lambda}^k, r_p^k)}. \tag{14}$$

Here s is an iteration counter for the number of approximate augmented Lagrangian minimizations which are performed for a fixed set of Lagrange multipliers $\boldsymbol{\lambda}^k$ and penalty term r_p^k. It is assumed that $\rho^s \to 1$ as the trust region radius $\Delta^s \to 0$ (i.e., for sufficiently small neighborhoods of the current iterate, the approximation $\tilde{\Phi}$ to Φ is arbitrarily accurate).

If $\rho^s \leq 0$, it means the approximation is bad and the trust region radius Δ^s has to be reduced. On the other hand, the radius has to be increased whenever $\rho^s \approx 1$ and $\|\mathbf{x}^{k,s} - \mathbf{x}^k\| = \Delta^s$. Notice that $\rho^s \gg 1$ also means that the approximation is bad, but the optimizer is moving in the right direction. Finally, if $\rho^s \in (0,1)$, then the radius could be kept unchanged, reduced, or increased depending on how close to 0 or 1 it is. In general, there is a lot of flexibility in the trust region updating. Many trust region methods make use of constant values in the updating procedure. Conn *et al.* (1988), make use of a procedure that changes Δ depending on the performance of the algorithm. The updating procedure used in the algorithm presented here is the following:

$$\begin{array}{lll} \Delta^{s+1} = \gamma_0 \Delta^s & \text{if} & \rho^s \leq 0.25, \\ \Delta^{s+1} = \Delta^s & \text{if} & 0.25 < \rho^s < 0.75, \\ \Delta^{s+1} = \gamma_2 \Delta^s & \text{if} & \rho^s \geq 0.75, \end{array} \tag{15}$$

where

$$\begin{array}{lll} \gamma_0 = 0.25, & & \\ \gamma_2 = 2 & \text{if} & \|\mathbf{x}^{k,s} - \mathbf{x}^k\| = \Delta^s, \\ \gamma_2 = 1 & \text{if} & \|\mathbf{x}^{k,s} - \mathbf{x}^k\| < \Delta^s. \end{array}$$

If $\rho^s \leq 0$ the iterate is rejected and the trust region radius is reduced according to (15).

The algorithm developed in this research is described in the next section and detailed in the flowchart in Figure 1. The inner loop of the algorithm is to solve approximately the problem (9), where F is formulated as the augmented Lagrangian function Φ, subject to the trust region bounds $\|\mathbf{x} - \mathbf{x}^k\| \leq \Delta^s$. Notice that if $\|\,.\,\|_\infty$ is used to define the trust region, the problem reduces to (9) in which the bounds are

$$\max\{x_l^L, x_l^k - \Delta^s\} \leq x_l \leq \min\{x_l^U, x_l^k + \Delta^s\}.$$

Let

$$\mathbf{c}(\mathbf{x}) = \left(\begin{array}{c} \mathbf{h}(\mathbf{x}) \\ \boldsymbol{\Psi}(\mathbf{x}) \end{array} \right).$$

An iteration of the algorithm (i.e., outer loop) is completed when the Lagrange multipliers or the penalty term are updated.

The algorithm finds designs which force $P(\nabla\Phi(\mathbf{x}^k, \boldsymbol{\lambda}^k, r_p^k))$ to zero as k increases. This progress is quantified by the parameter μ^k. In order to ensure convergence, the algorithm requires a reduction in the residual of the constraints after each Lagrangian iteration. This reduction is driven to zero by the parameter η^k. If a reduction is not accomplished after any Lagrangian iteration, the penalty parameter is increased keeping the Lagrange multipliers unchanged. The Lagrange multipliers are updated whenever a reduction in the residual of the constraints occurs. Notice that because of the test performed on the constraints' residual in Steps 3 and 5 of the algorithm (see next section), proper scaling of the constraints, as well as the objective function at the beginning of the iterative process is required.

The fidelity of the data used to build the response surface approximation is controlled through the integer variable Υ which ranges from $\Upsilon_{\min}$ to $\Upsilon_{\max}$. A value of $\Upsilon_{\max}$ means that exact values of the augmented Lagrangian Φ are used, while $\Upsilon_{\min}$ means that the lowest fidelity level is used. The fidelity of the data varies based on the fidelity of the design tool used to perform the analysis. Note that the algorithm developed in the next section updates Φ and $\nabla\Phi$ using high fidelity analyses after each approximate minimization (see Step 1). The response surface approximation generated in Step 10 of the algorithm requires that $\tilde{\Phi} = \Phi$ and $\nabla\tilde{\Phi} = \nabla\Phi$ at the current design. The fidelity variable Υ therefore controls the fidelity of the data sampled about the current design (i.e., Step 9) which is used to construct the second order response terms in Step 10.

Note that in the proposed algorithm the use of Υ is redundant in that the proposed algorithm requires zero and first order matching of Φ by the response surface approximation $\tilde{\Phi}$. The use of Υ as proposed does add robustness to the algorithm. In addition the use of Υ as proposed provides a general scheme for how one might use variable fidelity data in an approximate optimization strategy where the zero and first order matching are relaxed.

6 Description of the algorithm

Here the index k refers to a Lagrangian iteration of the algorithm (i.e., outer loop), and the index s refers to the number of approximate minimization steps (i.e., inner iterations) performed during a Lagrangian iteration.

Step 0. Initialization

Given an initial design vector $\mathbf{x}^0$, an initial vector of Lagrange multiplier estimates $\boldsymbol{\lambda}^0$, the penalty term r_p^0, and values for ϵ_1, ϵ_2, γ_1, Δ^0, η^0, Υ, and μ^0, set $k = 0$, $s = 0$. Evaluate $\Phi(\mathbf{x}^k,\boldsymbol{\lambda}^k, r_p^k)$ and $\nabla\Phi(\mathbf{x}^k,\boldsymbol{\lambda}^k, r_p^k)$. Go to Step 9.

Step 1. System Analysis and Global Sensitivity Equations

Evaluate $\Phi(\mathbf{x}^{k,s},\boldsymbol{\lambda}^k, r_p^k)$ and $\nabla\Phi(\mathbf{x}^{k,s},\boldsymbol{\lambda}^k, r_p^k)$.

Step 2. Compute the Trust Region Test

Compute the trust region test using (14). Modify the fidelity control Υ according to

$$\Upsilon = \begin{cases} \min\{\Upsilon_{\mathrm{old}} + 1, \Upsilon_{\max}\}, & \rho^s \le 0.25, \\ \Upsilon_{\mathrm{old}}, & 0.25 < \rho^s < 0.75, \\ \max\{\Upsilon_{\mathrm{old}} - 1, \Upsilon_{\min}\}, & \rho^s \ge 0.75. \end{cases}$$

If $\rho^s > 0.25$, update the trust region according to (15), set

$$\mathbf{x}^k := \mathbf{x}^{k,s}, \quad s := s + 1,$$

and go to Step 3, else, go to Step 8.

Step 3. Test of Convergence

If $\|P(\nabla\Phi(\mathbf{x}^k,\boldsymbol{\lambda}^k, r_p^k))\| \le \epsilon_1$ and $\|\mathbf{c}(\mathbf{x}^k)\|_\infty \le \epsilon_2$, stop. Otherwise, go to Step 4.

Step 4. Checking for Decrease in the Gradient of the Augmented Lagrangian

If $\|P(\nabla\Phi(\mathbf{x}^k,\boldsymbol{\lambda}^k, r_p^k))\| \le \mu^k$,then go to Step 5, else go to Step 9.

Step 5. Checking for Decrease in Constraint Violation If $\|\mathbf{c}(\mathbf{x}^k)\|_\infty \le \eta^k$ then go to Step 6, else, go to Step 7.

Step 6. Update Lagrange Multipliers

Update the Lagrange multipliers according to (7) and (8) and set

$$\begin{aligned} r_p^{k+1} &:= r_p^k, \\ \eta^{k+1} &:= \eta^k[\min\{1/r_p^{k+1}, \gamma_1\}]^{0.9}, \\ \mu^{k+1} &:= \mu^k \min\{1/r_p^{k+1}, 0.1\}, \end{aligned}$$

$k := k + 1$, $\mathbf{x}^k := \mathbf{x}^{k-1}$, $s = 0$, reinitialize the trust region radius, and go to Step 9.

Step 7. Increasing Penalty Term

Set

$$\begin{aligned} \boldsymbol{\lambda}^{k+1} &:= \boldsymbol{\lambda}^k, \\ r_p^{k+1} &:= 10 r_p^k, \\ \eta^{k+1} &:= \eta^0[\min\{1/r_p^{k+1}, \gamma_1\}]^{0.1}, \\ \mu^{k+1} &:= \mu^0 \min\{1/r_p^{k+1}, 0.1\}, \end{aligned}$$

$k := k + 1$, $\mathbf{x}^k := \mathbf{x}^{k-1}$, $s = 0$, reinitialize the trust region radius, and go to Step 9.

Step 8. Shrinking Move Limits

Update the trust region according to (15), $\mathbf{x}^k$ remains unchanged and $s := s + 1$. Go to Step 9.

Step 9. Data Base Generation

Set $\Upsilon_{\mathrm{old}} := \Upsilon$. Compute design data with fidelity Υ based on some sampling technique about the current iterate.

Step 10. Augmented Lagrangian Approximation

Build a response surface approximation $\tilde{\Phi}(\mathbf{x},\boldsymbol{\lambda}, r_p)$ of the augmented Lagrangian $\Phi(\mathbf{x},\boldsymbol{\lambda}, r_p)$ about the point $(\mathbf{x}^k, \boldsymbol{\lambda}^k)$ satisfying

$$\begin{aligned} \tilde{\Phi}(\mathbf{x}^k, \boldsymbol{\lambda}^k, r_p^k) &= \Phi(\mathbf{x}^k, \boldsymbol{\lambda}^k, r_p^k), \\ \nabla\tilde{\Phi}(\mathbf{x}^k, \boldsymbol{\lambda}^k, r_p^k) &= \nabla\Phi(\mathbf{x}^k, \boldsymbol{\lambda}^k, r_p^k). \end{aligned}$$

Step 11. Augmented Lagrangian Approximate Minimization

Find $\mathbf{x}^{k,s}$ such that

$$\tilde{\Phi}(\mathbf{x}^{k,s}, \boldsymbol{\lambda}^k, r_p^k) = \min_{\mathbf{x}\in\mathbf{X}} \tilde{\Phi}(\mathbf{x}, \boldsymbol{\lambda}^k, r_p^k),$$
$$\mathbf{x}^{k,s} \in \mathbf{X} = \{\mathbf{x} | \mathbf{x} \in S, \|\mathbf{x} - \mathbf{x}^k\| \le \Delta^s\}.$$

Go to Step 1.

A flow chart of the algorithm is shown in Figure 1.

In Steps 6 and 7 it is mentioned that the trust region radius has to be reinitialized. There are a number of ways in which this can be done. In our implementation the trust region radius is reinitialized as

$$\Delta^k = \kappa\|P(\nabla\Phi(\mathbf{x}^k, \boldsymbol{\lambda}^k, r_p^k))\|, \tag{16}$$

where $\kappa \ge 1$ is used. The trust region radius is selected in this way in order to avoid a trust region radius Δ that interferes with the steepest descent direction of the problem.

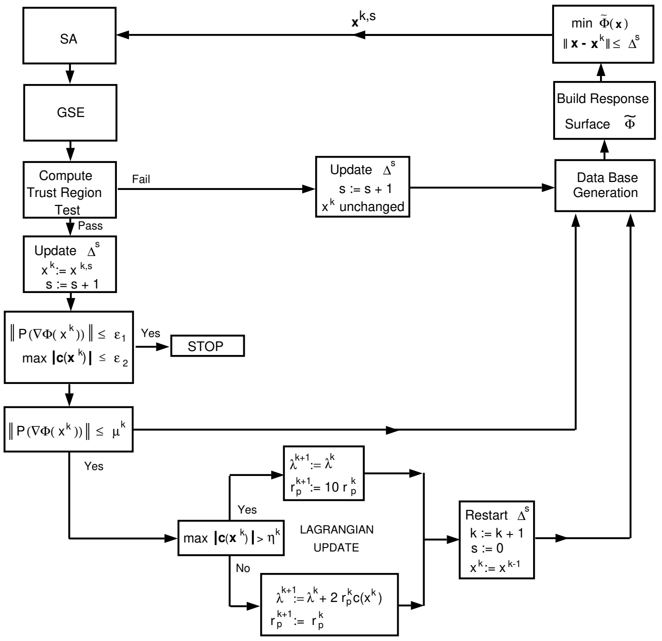

Figure 1: Algorithm flow chart.

7 Proof of convergence

In this section a formal proof of convergence is developed that guarantees the optimization process developed above (Figure 1) will converge to a Karush-Kuhn-Tucker point for the problem defined in equations (1) through (4). The algorithm defined in the previous section (Figure 1), is similar in nature to that developed in Conn *et al.* (1991). In the Conn *et al.* (1991) algorithm the optimizer interacts exclusively with the high fidelity model to either obtain exact second order information or to build a BFGS approximation of the second order information. In this research the optimizer interacts with a lower fidelity approximate model over the design space (i.e., response surface approximation). Both methods make use of trust region model management strategies that periodically query the high fidelity model to insure model accuracy and convergence of sequential minimizations. Convergence of the outer loop in Figure 1 (i.e., the Lagrangian iteration indexed by k) has been proven in Conn *et al.* (1991). Note that in this research a variation of the augmented Lagrangian formulation by Rockafellar (1973) is used (Equations 6 and 8). The careful reader will note that Rockafellar required convexity to prove global convergence of his scheme because the penalty parameter was fixed during the optimization process. Global convergence for the Rockafellar method can be obtained without assuming convexity when the penalty parameter is allowed to change as in the Conn *et al* (1991) proof of convergence for the outer loop. In this research an update of the penalty parameter is provided for in the algorithm (Step 7 - Description of the Algorithm). Therefore the assumptions required of the augmented Lagrangian formulation in the Conn *et al* proof of convergence of the outer loop are satisfied in the algorithm proposed in this research.

Therefore in order to prove convergence of the algorithm defined in the previous section (Figure 1), one needs to prove that the trust region managed sequence of approximate minimizations (i.e., the “inner” loop in Figure 1 indexed by iterations s) of the augmented Lagrangian subject to variable bounds will converge. A formal proof of convergence for a trust region managed sequence of approximate minimizations of a general objective function subject to variable bounds is detailed below. This proof applies to the trust region managed sequence of approximate minimizations of the augmented Lagrangian subject to variable bounds and therefore guarantees that the complete algorithm proposed above (Figure 1) will converge to a Karush-Kuhn-Tucker point for the problem defined in (1)–(4).

In proving convergence of the algorithm described in Figure 1 the following assumptions are made:

AS1 S has a nonempty interior and is compact.

AS2 f, g_i, and h_i are twice continuously differentiable on an open set containing S.

AS3 $\nabla\Phi(\mathbf{x})$ satisfies a Lipschitz condition.

AS4 The feasible set defined by (2)–(4) is nonempty.

AS5 $\rho^s \to 1$ as $\Delta^s \to 0$ or $\Upsilon \to \Upsilon_{\max}$.

AS6 $\mathbf{g}$ and $\mathbf{h}$ satisfy the Kuhn-Tucker constraint qualification or the modified Arrow-Hurwicz-Uzawa constraint qualification at any local minimum of the problem (1)–(4).

AS7 The Jacobian matrix of the active constraints has full rank at any limit point $\mathbf{x}^*$ of the sequences $\{\mathbf{x}^k\}$ considered.

Observe that AS1 implies the iterates $\{\mathbf{x}^k\}$ lie within a compact set. AS4 and AS6 guarantee that (13) can be satisfied, i.e., a KKT point for (1)–(4) exists. This set of assumptions suffices for the analysis developed in Conn *et al.* (1991) which proves the convergence of the outer loop (i.e., the Lagrangian iteration indexed by k) of the algorithm described in Figure 1.

Consider finding a solution to the problem

$$\min \quad \Phi(\mathbf{x}) \tag{17}$$

subject to

$$x_i^L \le x_i \le x_i^U, \quad i = 1 \dots n. \tag{18}$$

The algorithm that is going to be used is of the trust region type in which at each iteration a quadratic model of $\Phi(\mathbf{x})$ is approximately minimized subject to a trust region constraint and the bounds (18). Throughout, assume that $\Phi(\mathbf{x})$ is C^1 and the gradient $\nabla\Phi(\mathbf{x})$ satisfies a Lipschitz condition. The quadratic model is defined as

$$m(\mathbf{x}^s + \Delta\mathbf{x}) = \Phi(\mathbf{x}^s) + \nabla\Phi(\mathbf{x}^s)^T \Delta\mathbf{x} + \frac{1}{2}\Delta\mathbf{x}^T \mathbf{B}^s \Delta\mathbf{x}, \tag{19}$$

where $\nabla\Phi(\mathbf{x}^s)$ is the gradient of $\Phi(\mathbf{x})$ at $\mathbf{x}^s$, and $\mathbf{B}^s$ is an approximation to the Hessian of $\Phi(\mathbf{x})$ at $\mathbf{x}^s$. Note that m is a special case of the more general response surface approximation $\tilde{\Phi}$ from Step 10 of the algorithm.

The subproblem that is solved at each iteration of the algorithm can now be posed as

$$\min \quad m(\mathbf{x}^s + \Delta\mathbf{x}) \tag{20}$$

subject to the constraints

$$\|\Delta\mathbf{x}\| \le \Delta^s, \tag{21}$$

$$x_i^L \le x_i \le x_i^U, \quad i = 1 \dots n. \tag{22}$$

We define the projection operator componentwise by

$$(P[\mathbf{x}])_i = \begin{cases} x_i & \text{if} \quad x_i^L < x_i < x_i^U, \\ x_i^L & \text{if} \quad x_i \le x_i^L, \\ x_i^U & \text{if} \quad x_i \ge x_i^U, \end{cases} \tag{23}$$

where $P[\cdot]$ projects the point $\mathbf{x}^s$ onto the feasible region defined by (18).

The negative of $\nabla\Phi(\mathbf{x}^s)$ defines a descent direction for $\Phi(\mathbf{x})$ at the point $\mathbf{x}^s$. The projection of the ray $\mathbf{x}^s - t\nabla\Phi(\mathbf{x}^s)$, for $t \ge 0$, onto the feasible region (18) defines a polygonal ray $P[\mathbf{x}^s - t\nabla\Phi(\mathbf{x}^s)]$ for $t \ge 0$. This polygonal ray satisfies the non-decreasing property

$$\|P[\mathbf{x}^s - t_1\nabla\Phi(\mathbf{x}^s)] - \mathbf{x}^s\| \ge \|P[\mathbf{x}^s - t_2\nabla\Phi(\mathbf{x}^s)] - \mathbf{x}^s\|, \tag{24}$$

whenever $t_1 \ge t_2$. This property will be especially useful later.

Now we are in a position to define the piecewise quadratic model $q^s(t)$ as

$$q^s(t) = m(P[\mathbf{x}^s - t\nabla\Phi(\mathbf{x}^s)]). \tag{25}$$

We will denote as t_c^s the minimum value of t for which (25) is minimized subject to the trust region constraint

$$\|P[\mathbf{x}^s - t\nabla\Phi(\mathbf{x}^s)] - \mathbf{x}^s\| \le \nu\Delta^s, \tag{26}$$

where $\nu < 1$ is a positive constant. Note that the bounds are automatically accounted for in the projection operator $P[\cdot]$. The point given by

$$\mathbf{x}_c^s = P[\mathbf{x}^s - t_c^s \nabla\Phi(\mathbf{x}^s)]$$

is called the generalized Cauchy point.

The trial point $\mathbf{x}^s + \Delta\mathbf{x}^s$ has to be feasible and satisfy the two inequalities

$$\Phi(\mathbf{x}^s) - m(\mathbf{x}^s + \Delta\mathbf{x}^s) \ \ge\ \beta_1\left[\Phi(\mathbf{x}^s) - q^s(t_c^s)\right], \tag{27}$$

$$\|\Delta\mathbf{x}^s\| \ \le\ \beta_2\Delta^s. \tag{28}$$

Equations (27) and (28) require a reduction in the model that is a fraction of the model reduction at the generalized Cauchy point $\mathbf{x}_c^s$, inside an extended trust region with radius $\beta_2\Delta^s$. β_1 and β_2 are arbitrary positive constants satisfying

$$\beta_1 \in (0, 1], \qquad \beta_2 \ge \nu.$$

The norm that will be used in this paper is the infinity norm $\|\cdot\|_\infty$. Now we are ready to introduce the trust region algorithm used to solve (17) and (18). The algorithm depends on constants γ_1 and γ_2 satisfying

$$0 < \gamma_1 < 1 \le \gamma_2. \tag{29}$$

The algorithm is outlined below.

Given an initial feasible design $\mathbf{x}^0 \in R^n$ and trust region radius Δ^0, set $0 < \mu < \eta < 1$, and $s := 0$.

Execute the following steps until convergence.

1) Compute $\Phi(\mathbf{x}^s)$ and the model $m(\mathbf{x}^s + \Delta\mathbf{x})$.

2) Determine a solution $\Delta\mathbf{x}^s$ to problem (20), (21), (22).

3) Compute

$$\rho^s = \frac{\Phi(\mathbf{x}^s) - \Phi(\mathbf{x}^s + \Delta\mathbf{x}^s)}{\Phi(\mathbf{x}^s) - m(\mathbf{x}^s + \Delta\mathbf{x}^s)}. \tag{30}$$

4) If $\rho^s \ge \mu$,

$$\begin{array}{ll} \text{set} & \mathbf{x}^{s+1} := \mathbf{x}^s + \Delta\mathbf{x}^s, \\ \text{if } \rho^s > \eta & \Delta^s \in [\Delta^s, \gamma_2\Delta^s], \\ \text{if } \rho^s \in [\mu, \eta] & \Delta^s \in [\gamma_1\Delta^s, \gamma_2\Delta^s]. \end{array}$$

Otherwise

$$\mathbf{x}^{s+1} := \mathbf{x}^s,$$

$$\Delta^{s+1} \in [0, \gamma_1\Delta^s].$$

5) Update model $m(\mathbf{x}^{s+1} + \Delta\mathbf{x})$ and set $s := s + 1$.

In this algorithm, an iteration with $\rho^s \ge \mu$ is considered successful. An iteration not satisfying $\rho^s \ge \mu$ is an unsuccessful iteration.

The convergence analysis is based on the following extra assumptions:

AS8 The set $M \doteq \{\{\mathbf{x}|\Phi(\mathbf{x}) < \Phi(\mathbf{x}^0)\} \cap \{\mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U\}\}$, with $\mathbf{x}^0$ satisfying (18), is not empty.

AS9 $\|\mathbf{B}^s\| \leq \sigma_1$.

The first assumption means that the solution to (17) and (18) is nontrivial. The second assumption insures that the approximation of the Hessian remains uniformly bounded.

As a first step in the analysis, it is necessary to define a criterion for identifying a critical point. The following lemma provides such a criterion.

Lemma 1 *Let* $\mathbf{x}$ *be a feasible point. Then* $\mathbf{x}$ *is a stationary point for problem (17), (18) if and only if*

$$\mathbf{x}(t) = P[\mathbf{x} - t\nabla\Phi(\mathbf{x})],$$

for all $t \geq 0$.

Proof.

A stationary point $\mathbf{x}^*$ for problem (17), (18) is characterized by

$$\begin{aligned} \frac{\partial\Phi(\mathbf{x}^*)}{\partial x_i} &= 0 \quad \text{if} \quad x_i^L < x_i < x_i^U, \\ \frac{\partial\Phi(\mathbf{x}^*)}{\partial x_i} &\geq 0 \quad \text{if} \quad x_i = x_i^L, \\ \frac{\partial\Phi(\mathbf{x}^*)}{\partial x_i} &\leq 0 \quad \text{if} \quad x_i = x_i^U. \end{aligned} \tag{31}$$

Necessity. Using (31) and the definition of $P[\cdot]$, it is easily verified that a stationary point $\mathbf{x}$ satisfies $\mathbf{x} = P[\mathbf{x} - t\nabla\Phi(\mathbf{x})]$.

Sufficiency. Let $x_i^L < x_i < x_i^U$. Then for all sufficiently small $t > 0$,

$$\begin{aligned} x_i &= P[x_i - t(\nabla\Phi(\mathbf{x}))_i] \\ &= x_i - t(\nabla\Phi(\mathbf{x}))_i \end{aligned}$$

which implies

$$(\nabla\Phi(\mathbf{x}))_i = 0.$$

For $x_i = x_i^L$, $x_i^L = P[x_i^L - t(\nabla\Phi(\mathbf{x}))_i]$ $\forall t \geq 0 \Rightarrow (\nabla\Phi(\mathbf{x}))_i \geq 0$, and similarly for $x_i = x_i^U$, $x_i^U = P[x_i^U - t(\nabla\Phi(\mathbf{x}))_i]$ $\forall t \geq 0 \Rightarrow (\nabla\Phi(\mathbf{x}))_i \leq 0$. □

Now, let $\mathbf{x}^s$ be feasible and

$$\mathbf{d}^s(t) = P[\mathbf{x}^s - t\nabla\Phi(\mathbf{x}^s)] - \mathbf{x}^s. \tag{32}$$

Note that $\mathbf{d}^s(t) = 0$ means that $\mathbf{x}^s$ is a stationary point by Lemma 1. Another way to classify a stationary point is by means of

$$v^s = \|P[\mathbf{x}^s - \nabla\Phi(\mathbf{x}^s)] - \mathbf{x}^s\| = \|\mathbf{d}^s(1)\| = 0. \tag{33}$$

The first step for proving global convergence of the algorithm is to prove that after a successful iteration, a reduction in the function $\Phi(\mathbf{x})$ is obtained, namely

$$\Phi(\mathbf{x}^s) - \Phi(\mathbf{x}^s + \Delta\mathbf{x}^s) \geq c_1(v^s)^2 \min\left\{\Delta^s, \frac{(v^s)^2}{c_2}\right\}, \tag{34}$$

for some constants c_1, $c_2 > 0$.

To show (34) we will follow the proof given by Conn *et al.* (1988).

Before proceeding with the proof, let us define the set $\mathcal{I}$ of the indices of the active variables as

$$\mathcal{I} = \{i \mid |x_i = x_i^L \text{ or } x_i = x_i^U\}.$$

Because of the non-decreasing property (24) of the polygonal ray, it follows that

$$\mathcal{I}(\mathbf{x}^s + \mathbf{d}^s(t_1)) \subseteq \mathcal{I}(\mathbf{x}^s + \mathbf{d}^s(t_2)), \quad \text{for } t_2 \geq t_1.$$

We also define the projected gradient as

$$[\nabla_p\Phi(\mathbf{x}^s + \mathbf{d}^s(t))]_i = \begin{cases} [\nabla\Phi(\mathbf{x}^s + \mathbf{d}^s(t))]_i & \text{if } i \notin \mathcal{I}(\mathbf{x}^s + \mathbf{d}^s(t)), \\ 0 & \text{otherwise.} \end{cases} \tag{35}$$

Lemma 2 (Conn *et al.* (1988), Lemma 3) *Assume* $\mathbf{x}^s$ *is feasible and that* $v^s > 0$. *Then if*

$$c_3 \doteq \max\{1, \max_{x \in M} \|\nabla\Phi(\mathbf{x})\|\},$$

we obtain that

$$\|\nabla_p\Phi(\mathbf{x}^s + \mathbf{d}^s(t^s_{(1)}))\| \geq \frac{1}{2}v^s,$$

where

$$t^s_{(1)} = \frac{v^s}{2c_3}.$$

The next lemma establishes the decreasing property for the piecewise quadratic model.

Lemma 3 (Conn *et al.* (1988), Lemma 4) *Assume* $\mathbf{x}^s$ *is feasible and let*

$$\alpha^s \doteq \|\nabla_p\Phi(\mathbf{x}^s + \mathbf{d}^s(t^s_{(3)}))\| > 0.$$

If $\mathcal{T}$ *corresponds to the set of points in* $[0, t^s_{(3)}]$ *at which the piecewise quadratic model* $q^s(t)$ *is differentiable, then*

$$\frac{d}{dt}q^s(t) \leq -(\alpha^s)^2 + 2t^s_{(3)}\|\nabla\Phi(\mathbf{x}^s)\|^2\|B^s\|,$$

for all $t \in \mathcal{T}$. Furthermore

$$q^s(t) \le \Phi(\mathbf{x}^s) - \frac{1}{2}(\alpha^s)^2 t,$$

for all $t \in [0, t^s_{(4)}]$ where

$$t^s_{(4)} \doteq \min\left\{ t^s_{(3)}, \frac{(\alpha^s)^2}{4c_3^2(1+\sigma_1)} \right\}.$$

The next lemma shows that the algorithm produces a reduction in the model $m(\mathbf{x}^s + \Delta\mathbf{x})$ and if the iteration is successful, a reduction in the function $\Phi(\mathbf{x})$.

Lemma 4 *Assume $\mathbf{x}^s$ is feasible and that*

$$v^s \doteq \|P[\mathbf{x}^s - \nabla\Phi(\mathbf{x}^s)] - \mathbf{x}^s\| > 0.$$

Then

$$\Phi(\mathbf{x}^s) - m(\mathbf{x}^s + \Delta\mathbf{x}^s) \ge \frac{1}{2} c_4 (v^s)^2 \min\left\{ \Delta^s, \frac{(v^s)^2}{16c_3(\sigma_1+1)} \right\},$$

where

$$c_4 \doteq \frac{\beta_1 \nu}{4c_3}.$$

If the iteration is successful then

$$\Phi(\mathbf{x}^s) - \Phi(\mathbf{x}^s + \Delta\mathbf{x}^s) \ge \frac{1}{2} c_4 \mu (v^s)^2 \min\left\{ \Delta^s, \frac{(v^s)^2}{16c_3(\sigma_1+1)} \right\}.$$

Proof.

If in Lemma 3, $t^s_{(3)}$ is replaced by $t^s_{(1)}$ given in Lemma 2, then $\alpha^s \ge v^s/2$ and the piecewise quadratic model satisfies

$$q^s(t) \le \Phi(\mathbf{x}^s) - \frac{(v^s)^2 t}{8},$$

where t has the bound

$$t^s_{(5)} = \min\left\{ \frac{v^s}{2c_3}, \frac{(v^s)^2}{16c_3^2(1+\sigma_1)} \right\} = \frac{(v^s)^2}{16c_3^2(1+\sigma_1)},$$

since $v^s \le c_3$.

Now, if $\|\mathbf{d}^s(t^s_{(5)})\| \le \nu\Delta^s$ then

$$\Phi(\mathbf{x}^s) - q^s(t^s_c) \ge \Phi(\mathbf{x}^s) - q^s(t^s_{(5)}) \ge \frac{(v^s)^2 t^s_{(5)}}{8},$$

or by (27) and since $\nu < 1$,

$$\Phi(\mathbf{x}^s) - m(\mathbf{x}^s + \Delta\mathbf{x}^s) \ge \frac{\beta_1 (v^s)^2 t^s_{(5)}}{8} \ge \frac{\beta_1 \nu (v^s)^2 t^s_{(5)}}{8}. \tag{36}$$

Now assume that $\|\mathbf{d}^s(t^s_{(5)})\| > \|\mathbf{d}^s(t^s_c)\| = \nu\Delta^s$. Since

$$\left\| \mathbf{d}^s\left(\frac{\nu\Delta^s}{c_3}\right) \right\| \le \frac{\nu\Delta^s}{c_3} \|\nabla\Phi(\mathbf{x}^s)\| \le \nu\Delta^s,$$

it follows that $t^s_{(5)} > t^s_c \ge (\nu\Delta^s)/c_3$, and in this case

$$\Phi(\mathbf{x}^s) - m(\mathbf{x}^s + \Delta\mathbf{x}^s) \ge \frac{\beta_1 (v^s)^2 t^s_c}{8} \ge \frac{\beta_1 \nu (v^s)^2 \Delta^s}{8c_3}. \tag{37}$$

From (36) and (37) we obtain

$$\Phi(\mathbf{x}^s) - m(\mathbf{x}^s + \Delta\mathbf{x}^s) \ge \frac{1}{2} c_4 (v^s)^2 \min\left\{ \Delta^s, \frac{(v^s)^2}{16c_3(\sigma_1+1)} \right\}.$$

If the iteration is successful, then (30) and the fact that $\rho^s \ge \mu$ lead to

$$\Phi(\mathbf{x}^s) - \Phi(\mathbf{x}^s + \Delta\mathbf{x}^s) \ge \frac{1}{2} c_4 \mu (v^s)^2 \min\left\{ \Delta^s, \frac{(v^s)^2}{16c_3(\sigma_1+1)} \right\}.$$

□

Theorem 1 *If AS1–AS9 hold then*

$$\lim_{s\to\infty} v^s = 0.$$

Proof.

We have to prove that

$$\limsup_{s\to\infty} v^s = 0.$$

Let us assume that there exists an $\epsilon > 0$ such that $v^s > \epsilon$ for infinitely many s. Let T be the set of these indices.

Now we will prove that

$$\sum_{s\in T} \Delta^s < +\infty. \tag{38}$$

By Lemma 4, a successful step results in a reduction

$$\Phi(\mathbf{x}^s) - \Phi(\mathbf{x}^s + \Delta\mathbf{x}^s) \ge \min\left\{ \frac{c_1}{c_2}\epsilon^4, c_1\epsilon^2\Delta^s \right\}$$

for $s \in T$. Split the sum in (38) into three parts, Σ_1, Σ_2, Σ_3. Let Σ_1, Σ_2 contain those terms corresponding to successful steps for which the minimum above is $c_1\epsilon^4/c_2$, $c_1\epsilon^2\Delta^s$, respectively, and let Σ_3 correspond to unsuccessful steps. Since $\Phi(\mathbf{x})$ is bounded below on M, the reduction in Φ must be finite, so Σ_1 contains only finitely many terms. Similarly Σ_2 (which might have infinitely many terms) must be finite. Let $\widetilde{\Delta} = \max \Delta^s$ over the terms in Σ_1 and Σ_2. The *pth* unsuccessful step corresponds to Δ^{s_p} and satisfies

$$\Delta^{s_p} \le \gamma_1^p \widetilde{\Delta},$$

so

$$\Sigma_3 = \sum_p \Delta^{s_p} \le \widetilde{\Delta}(\gamma_1 + \gamma_1^2 + \cdots) = \widetilde{\Delta}\frac{\gamma_1}{1-\gamma_1} < \infty.$$

Furthermore, the convergence of (38) implies

$$\Delta^s \to 0(s \in T) \Rightarrow \Delta \mathbf{x}^s \to 0(s \in T),$$

since $\|\Delta \mathbf{x}^s\| \leq \beta_2 \Delta^s$. Now

$$\Phi(\mathbf{x}^s + \Delta \mathbf{x}^s) - \Phi(\mathbf{x}^s) = \nabla \Phi(\mathbf{x}^s) \Delta \mathbf{x}^s + o(\|\Delta \mathbf{x}^s\|)$$

and

$$m(\mathbf{x}^s + \Delta \mathbf{x}^s) - \Phi(\mathbf{x}^s) = \nabla \Phi(\mathbf{x}^s) \Delta \mathbf{x}^s + o(\|\Delta \mathbf{x}^s\|)$$

together with $v^s > \epsilon$ for $s \in T$, imply that $\rho^s \to 1$ for $s \in T$.

Now, the updating rules for the algorithm show that Δ^s is not decreasing if $\rho^s > \eta$, thus $\{\Delta^s\}$ cannot converge to zero for $s \in T$, which is a contradiction. Therefore

$$\limsup_{s \to \infty} v^s = 0.$$

□

Finally, observe that since a stationary point corresponds to $v^s = 0$, Theorem 1 proves that the trust region algorithm converges to a stationary point of (17), (18).

8 Variable fidelity data and CSSO-MDO

The proof of convergence developed in the previous section is directly applicable to the algorithm proposed here (Figure 1). Note that the proof of convergence developed in the previous section places no restrictions on how the design data is generated and limited restrictions on the fidelity of that data. In preliminary implementation studies (Rodríguez *et al.* (1997)), high fidelity models were queried at each design site specified by a statistical sampling plan in order to construct response surface approximations of the augmented Lagrangian components. In this research we elect to use the principles of the concurrent subspace optimization (CSSO) of Wujek *et al.* (1996, 1997) to generate design data (Step 9 of the algorithm) for multidisciplinary systems at a much lower cost.

The concurrent subspace optimization (CSSO) approach of Wujek *et al.* (1996, 1997) provides for the temporary decoupling of the disciplinary analyses during subspace optimizations as first proposed in Sobieszczanski-Sobieski (1988). The subspace optimizers in the Wujek *et al.* CSSO algorithm each solve the same system level optimization problem. The optimizations differ in that each discipline (i.e., subspace) optimizes the system level problem with respect to a local set of design variables allocated to the subspace. Evaluations of the system states (i.e., performance metrics) at the subspace level does not involve an iterative solution strategy. This is accomplished by providing the disciplinary designers with linear response representations of the other discipline states based on sensitivities generated from the global sensitivity equations (GSE) of Sobieszczanski-Sobieski (1990). Therefore nonlocal states are linearly approximated and local states are evaluated using the high fidelity local analysis tools available in the discipline. The design data generated during a subspace optimization is therefore of variable fidelity. Some of the data is generated by the high fidelity disciplinary analysis tools but based on linearized inputs of the other disciplines. Other data is simply the state prediction provided by the linearized response representations of non-local disciplinary states. This variable fidelity data is archived for each design site visited during a given subspace optimization. The subspace optimizers therefore provide the approximate response sampling strategy. The CSSO sampling strategy is optimization based and provides for design sampling in descent directions, where the sampling sites are feasible as measured by the approximations available within the subspaces.

In this research the subspace optimization formulation of Wujek *et al.* (1996,1997) is used to generate variable fidelity data (see Step 9—Description of the Algorithm) for use in constructing quadratic response surface approximations of the objective function and constraints. These response surface approximations are in turn used to construct a cumulative response surface approximation (Rodríguez *et al.* 1997) of the augmented Lagrangian (see Step 10—Description of the Algorithm). This strategy avoids having to perform costly iterative system analyses with high fidelity disciplinary analysis tools in order to construct a response surface approximation of the augmented Lagrangian. The use of variable fidelity data to construct response surface approximations is predicated on the ability of the trust region model management strategy to effectively manage the convergence of the approximate optimization process. The results presented in the next section show that response surface approximations constructed from variable fidelity data generated during concurrent subspace optimizations (CSSO) can be effectively managed by the trust region model management approach developed in this research. This implementation of the proposed algorithm results in a new variant of CSSO which is provably convergent under the restrictions AS1–AS9 mentioned earlier.

9 Numerical results

The algorithm was applied to two MDO test problems, each with a different dimension and degree of nonlinearity. Second order response surfaces of the objective and constraints based on data obtained from the concurrent subspace optimizations were built. These individual approximations were used to form a cumulative response surface approximation (Rodríguez *et al.* 1997) of the augmented Lagrangian in the coordination procedure of the algorithm. The SQP routine from the IMSL library was used in the augmented Lagrangian approximate minimization (i.e., Step 11).

9.1 Aircraft concept sizing (ACS)

This test problem was originally developed in Sellar *et al.* (1996a). The goal is to minimize the total weight of the aircraft subject to constraints on range and stall velocity. Eight fixed parameters and seven design variables compose the inputs to the analysis which computes six states. The system analysis is comprised of three disciplines: aerodynamics, weights, and performance which interact in a feed-forward fashion as shown in Figure 2.

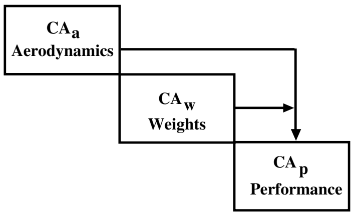

Figure 2: Dependency diagram for the ACS problem.

The aerodynamics contributing analysis (CA) calculates the wetted area S_w and the lift to drag ratio L/D. The value of the objective, total weight, is computed by the weights CA which also calculates the empty weight of the aircraft. The constraints, range and stall velocity, are calculated by the performance CA. This CA requires the lift to drag ratio, L/D, and the total weight, W_{total}, as inputs.

The optimization of the MDO formulation of the aircraft concept sizing problem can be posed as

$$\min \quad W_{\text{total}} = W_{\text{empty}} + W_{\text{fuel}} + W_{\text{payload}}$$

subject to

$$\frac{Range}{Range_{\text{required}}} - 1.0 \geq 0.0,$$
$$1.0 - \frac{V_{\text{stall}}}{V_{\text{stall}_{\text{required}}}} \geq 0.0.$$

The constraint boundaries are set at $Range_{\text{required}} =$ 565 miles and $V_{\text{stall}} = 70$ ft/s with the design variable bounds for the problem given in Table 1.

Design Variable	Lower Bound	Upper Bound	Starting Point
Aspect Ratio	5.0	9.0	7.0
S_{wing} (ft^2)	100.0	300.0	200.0
l_f (ft)	20.0	30.0	22.0
D_f (ft)	4.0	5.0	4.2
ρ_{cruise} (slugs/ft^3)	0.0019	0.002378	0.0021
V_{cruise} (ft/s)	200.0	300.0	250.0
W_{fuel} (lb)	100.0	400.0	200.0

Table 1: Design variable bounds for the ACS problem.

The problem has two minima which are shown in Table 2. At the global optimum five design variables are at their bounds and both constraints are active.

Design Variable	Global Optimum	Local Optimum
Aspect Ratio	5.0	7.0
S_{wing} (ft^2)	177.9	182.0
l_f (ft)	20.0	20.0
D_f (ft)	4.0	4.0
ρ_{cruise} (slugs/ft^3)	0.0019	0.0019
V_{cruise} (ft/s)	200.0	200.0
W_{fuel} (lb)	145.1	127.0
W_{total} (lb)	1762	1802

Table 2: Optimum solutions for the ACS problem.

The optimization of this problem was carried out using the CSSO data base generation approach in Step 9 of the algorithm. The starting point for this problem is given in the third column of Table 1. The algorithm converged to the global optimum given in Table 2 in four approximate minimizations as observed in the convergence plot of Figure 3. After the very first iteration, the optimizer moves to a point that is slightly infeasible, causing the objective to increase in value. The set of active variables is identified after the second iteration of the algorithm. In the remaining two iterations the algorithm follows the constraints until satisfying the KKT optimality conditions. Note that only five high fidelity analyses of the Lagrangian Φ were required for convergence, and that the gradient $\nabla\Phi$ was calculated using the Global Sensitivity Equations (GSE) which avoid high fidelity coupled analyses.

A comparison of the performance of the algorithm with the MATLAB function *constr* was conducted. The *constr* function is an implementation of the sequential quadratic programming (SQP) algorithm (see Grace (1992)). The starting point given in Table 1 was supplied to the MATLAB function *constr*, which required 52 high fidelity anal-

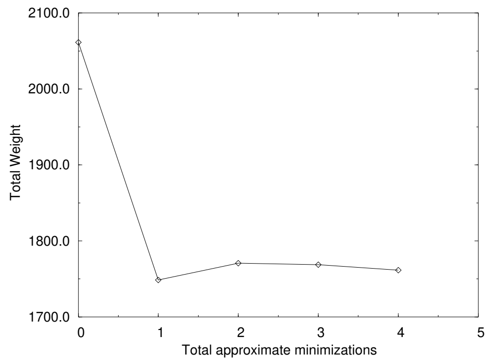

Figure 3: History of convergence for the ACS problem.

yses and 18 SQP iterations to converge to the global optimum given in Table 2. Note that this total includes a high fidelity analyses for each SQP iterate and the number of high fidelity analyses required in the SQP line searches. It does not count high fidelity analyses required for gradient calculations. Use of the algorithm developed here resulted in a savings of more than 90% when comparing the number of high fidelity analyses required for convergence.

9.2 Autonomous hovercraft

The design of an autonomous hovercraft (AHC), shown in Figure 4, was first presented in Sellar *et al.* (1996) and a full description can be found in Sellar (1997). This problem provides a further increase in size and complexity over the aircraft concept sizing problem and involves 11 design variables and the calculation of 50 states. The physical system consists of an engine, rotor, and payload. The rotor is comprised of two rectangular lifting surfaces located on opposite ends of a hollow, circular shaft. The system is to operate such that the motor speed (RPM) provides a thrust-to-weight ratio of one, imposing hover conditions.

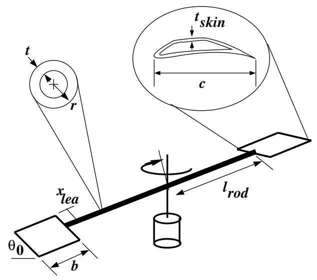

Figure 4: Autonomous hovercraft system.

The system analysis is comprised of four contributing analysis, three of which interact in a complex coupled fashion as illustrated in the dependency diagram of Figure 5. The aerodynamics CA (CA_a) calculates the aerodynamic loads on the lifting surfaces and approximates the distributed drag force along the rod while 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), the motor RPM and thrust as inputs. The shaft deformation is supplied by the structures CA (CA_s). All the quantities calculated by this CA, which also include the axial and shear stresses at the hub and the deflection of the lifting surfaces, are a function of the aerodynamic loads, thus subjecting these CAs to static aeroelastic coupling. The propulsion/performance CA (CA_p) calculates the thrust and torque necessary to spin the motor based on the loads supplied by CA_a. RPM is calculated by explicitly imposing that the total weight equals the thrust, determining the weight of the motor to achieve this total weight, and calculating the power and resulting RPM available from that size motor. At this point the formulation used here differs from that in Sellar *et al.* (1996) where the RPM is updated in a heuristic manner. The fourth CA, structural dynamics (CA_d), calculates the first natural frequencies of the rotor in bending and torsion. This CA is completely uncoupled from the other CAs as it requires no states as inputs.

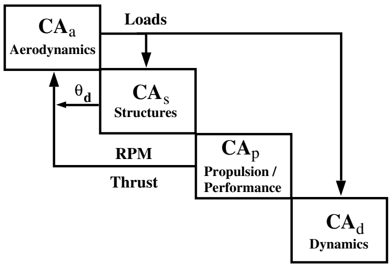

Figure 5: AHS dependency diagram.

The goal of the optimization of this system is to minimize the empty weight of the hovercraft subject to constraints on the Von Misses stress due to in-plane and normal forces in the rod, the first natural frequencies of the rod, the Mach number at the tip, and the hovercraft range. The optimization problem is posed as:

$$\min \quad W_{empty} = W_{wing} + W_{rod} + W_{fuel} + W_{motor}$$

subject to

$$g_1 = 1.0 - \frac{\sigma_N}{\sigma_{all}} \geq 0.0,$$

$$g_2 = 1.0 - \frac{\sigma_T}{\sigma_{all}} \geq 0.0,$$

$$g_3 = \frac{\omega_b}{k \cdot RPM} - 1.0 \geq 0.0,$$

$$g_4 = \frac{\omega_t}{k \cdot RPM} - 1.0 \geq 0.0,$$

$$g_5 = \frac{M_{tip_{all}}}{M_{tip}} \geq 0.0,$$

$$g_6 = \frac{E}{E_{req}} - 1.0 \geq 0.0.$$

where $\sigma_{all} = 14000\ psi$, $k = 1.5$, $M_{tip_{all}} = 0.8$ and $E_{req} = 2\ hrs$. The global optimum for this problem was reported in Sellar (1997) to be $W_{empty} = 67.9\ lbs$, for which 7 design variables are at their bounds and the hovercraft range constraint, g_6, is active.

Design Variable	Starting Design	Optimum Design
t (in)	0.5	0.15
r (in)	2.0	1.20
l_{rod} (in)	60.0	39.92
c (in)	18.0	6.00
b (in)	24.0	42.0
$\theta_\circ$ (deg)	15.0	10.66
t/c	0.2	0.05
x/c	0.0	0.1507
$\%cam$	0.0	0.035
t_{skin} (in)	0.25	0.05
W_{fuel} (lb)	50.0	19.981
W_{empty} (lb)	394.2	67.92

Table 3: Starting and optimum designs for the AHC problem.

The optimization of this problem was carried out via the algorithm described in previous sections, with CSSO used to generate the data which is used in building the response surface approximation of the augmented Lagrangian. The starting point and the optimum found are listed in Table 3.

Figure 6 shows the history of convergence of the objective. KKT conditions were satisfied at the optimum point. It was observed that the solution is insensitive to the design variable x/c as reported in Sellar (1997). This is because at the optimum point there is only one constraint active (i.e., g_6) for which its sensitivity as well as the objective's sensitivity with respect to this particular design variable is very low. For this reason, it is expected that designs will be obtained that are slightly different and for which optimality conditions are still satisfied. In this case twenty three high fidelity analyses of Φ were required for convergence. The gradient $\nabla\Phi$ was calculated using the Global Sensitivity Equations (GSE) which avoid high fidelity coupled analyses.

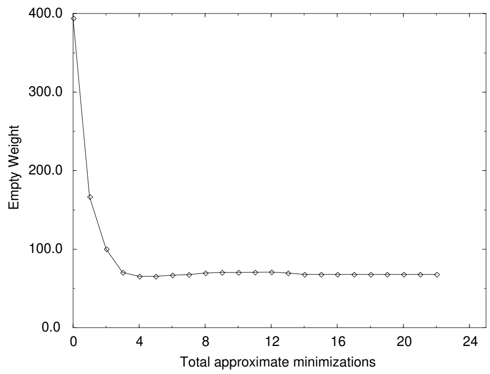

Figure 6: AHS objective convergence plot.

A comparison of the performance of the algorithm with the MATLAB function *constr* was conducted. The starting point given in Table 3 was supplied to the MATLAB function *constr*, which required 28 high fidelity analyses and 14 SQP iterations to find the optimum given in Table 3. Note that this total includes a high fidelity analyses for each SQP iterate and the number of high fidelity analyses required in the SQP line searches. It does not count high fidelity analyses required for gradient calculations. Use of the trust region augmented Lagrangian algorithm developed here resulted in a savings of more than 15% when comparing the number of high fidelity analyses required for convergence.

10 Conclusions

The focus in this research has been to prove that a recently developed constrained approximate multidisciplinary optimization strategy is assured of converging to a Karush-Kuhn-Tucker point for the problem defined in (1)–(4). The algorithm being studied sequentially optimizes a *variable fidelity response surface based approximation* of the augmented Lagrangian subject to a trust region constraint and model management strategy. A formal proof of convergence for this response surface approximation based optimization algorithm has been given. In an effort to augment the formal proof of convergence, numerical studies in application to two multidisciplinary design optimization (MDO) test problems confirm that the algorithm is effective in managing the convergence of approximate optimizations using variable fidelity data. Convergence to a Karush-Kuhn-Tucker (KKT) point is observed for both test problems. More importantly the numerical studies confirm that response surface approximations constructed from variable fidelity data generated during concurrent subspace optimizations (CSSOs) can be effectively managed by the trust region model management strategy. Extending application of the algorithm to the class of MDO problems and using variable fidelity data generated by CSSO are important contributions of this research. Note that the preliminary implementation studies of Rodríguez *et al.* (1997) were conducted on single level optimization problems using high fidelity design models to construct response surface approximations based on conventional statistical response sampling techniques. Whereas in the present work the CSSO generated design data is of variable fidelity and the sampling strategy is optimization based. Note that the cost of generating variable fidelity data is considerably less expensive computationally than querying a high fidelity model to construct response surface approximations, which is particularly important in MDO applications.

Acknowledgment

This research effort was supported in part by the following grants and contracts: NASA Research grant Awards NAG-1-1561 and NAG-1-1562, NSF Grants DMI94-57179 and DMS-9625968, and the support from the Venezuelan government through CONICIT. The authors thank Natalia Alexandrov and Raphael Haftka for insightful comments.

References

Alexandrov, N.; Dennis J. E. 1997: A Class of General Trust-Region Multilevel Algorithms for Systems of Nonlinear Equations and Equality Constrained Optimization: Global Convergence Theory. Submitted to *SIAM Journal on Optimization*.

Alexandrov, N. 1996: Robustness Properties of a Trust Region Framework for Managing Approximations in Engineering Optimization. *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium* (held at Bellevue, WA). Paper 96-4102, pp. 1056–1059.

Balabanov, V.; Kaufman, M.; Knill, D. L.; Haim, D.; Golovidov, O.; Giunta, A. A.; Haftka, R. T.; Grossman, B.; Mason, W. H.; Watson, L. T. 1996: Dependence of Optimal Structural Weight on Aerodynamic Shape for a High Speed Civil Transport. *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium* (held at Bellevue WA). Paper 96-4046, pp. 599–613.

Bloebaum, C. L.; Hong, W., Peck, A. 1994: Improved Moved Limit Strategy for Approximate Optimization. *Proceedings of the Fifth AIAA/USAF/NASA/ISSMO Symposium* (held at Panama City, FL). Paper 94-4337-CP, pp. 843–850.

Burgee, S.; Giunta, A.; Balabanov, V.; Grossman, B.; Mason, W. H.; Narducci, R.; Haftka R. T.; Watson L. T. 1996: A coarse grained parallel variable-complexity multidisciplinary optimization paradigm. *Internat. J. Supercomputer Appl. High Performance Comput.* **10**, 269–299.

Chen, W.; Allen, J. K.; Schrage, D. P.; Mistree, F. 1996: Statistical Experimentation for Affordable Concurrent Design. *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium* (held at Bellevue, WA). Paper 96-4085, pp. 921–930.

Chen, W.; Simpson, T. W.; Allen, J. K.; Mistree, F. 1996: Using Design Capability Indices to Satisfy Sets of Design Requirements. *Proceedings of the 1996 ASME Design Engineering Technical Conference and Computers in Engineering Conference* (held at Irvine, CA). Paper 96-DETC / DAC-1090, CD-ROM Proceedings, ISBN 0-7918-1234-4.

Chen, W.; Tsui, K. L.; Allen, J. K.; Mistree, F. 1995: Integration of the Response Surface Methodology with the Compromise Decision Support Problem in Developing a General Robust Design Procedure. *Proceedings of the 1995 ASME Design Engineering Technical Conferences, Advances in Design Automation* (held at Boston, Massachusetts). ASME DE-Vol. 82, pp. 485–492, (eds. S. Azarma et al.).

Chen, T. Y. 1993: Calculation of the Move Limits for the Sequential Linear Programming Method. *International Journal for Numerical Methods in Engineering* **36**, 2661–2679.

Conn, A. R.; Gould, N. I. M.; Toint, Ph. L. 1988: Global Convergence of a Class of Trust Region Algorithms for Optimization with Simple Bounds. *SIAM J. Numer. Anal.* **25**, No. 2, 433–464.

Conn, A. R.; Gould, N. I. M.; Toint, Ph. L. 1988b: Testing a Class of Methods for Solving Minimization Problems with Simple Bounds on the Variables. *Mathematics of Computation* **50** (182), 399–430.

Conn, A. R.; Gould, N. I. M.; Toint, Ph. L. 1991: A Globally Convergent Augmented Lagrangian Algorithm for Optimization with General Constraints and Simple Bounds. *SIAM J. Numer. Anal.* **28**, No. 2, 545–572.

Conn, A. R.; Gould, N. I. M.; Toint, Ph. L. 1992: *LANCELOT, a FORTRAN Package for*

Large-Scale Nonlinear Optimization. Springer Series in Computational Mathematics 17, Springer-Verlag, Berlin.

Dennis, J.E.; Torczon, T. 1996: Approximation Model Management for Optimization. *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium* (held at Bellevue, WA). Paper 96-4046, pp. 1044–1046.

Fadel, G. M.; Riley, M. F.; Barthelemy, J. F. M. 1990: Two Point Exponential Approximation Method for Structural Optimization. *Structural Optimization* **2**, 117–124.

Giunta, A.; Balabanov, V.; Kaufman, M.; Burgee, S.; Grossman, B.; Haftka, R. T.; Mason, W. H.; Watson, L. T. 1996: Variable-complexity response surface design of an HSCT configuration. In: N. M. Alexandrov and M. Y. Hussaini (eds), *Multidisciplinary Design Optimization*, pp. 348–367. Philadelphia: SIAM.

Giunta, A. A.; Dudley, J. M.; Narducci, R.; Grossman, B.; Haftka, R. T.; Mason, W. H.; Watson, L. T. 1994: Noisy Aerodynamic Response and Smooth Approximations in HSCT Design. *Proceedings of the Fifth AIAA/USAF/NASA/ISSMO Symposium* (held at Panama City, FL). Paper 94-4376, pp. 1117–1128.

Grace, A. 1992: *Optimization Toolbox for use with MATHLAB*. The Math Works, Inc., Natick, MA.

Grignon, P.; Venkatesh V.; Fadel, G. M. 1994: Fuzzy Moved Limit Evaluation in Structural Optimization. *Fifth AIAA/USAF/NASA/ISSMO Conference on Multidisciplinary Analysis and Optimization* (held at Panama City, FL). Paper 94-4281.

Hestenes M. R. 1969: Multiplier and Gradient Methods. *J. Optimization Theory and Applications* **4**, 303–320.

Himmelblau, D. M. 1972: *Applied Nonlinear Programming*. McGraw Hill, New York.

IMSL MATH/LIBRARY. *Fortran Subroutines for Mathematical Applications*, Vol III. Houston, TX.

Koch, P. N.; Barlow, A.; Allen, J. K.; Mistree, F. 1996: Configuring Turbine Propulsion Systems Using Robust Concept Exploration. *Proceedings of the 1996 ASME Design Engineering Technical Conference and Computers in Engineering Conference* (held at Irvine, CA). Paper 96-DETC / DAC-1472, CD-ROM Proceedings, ISBN 0-7918-1234-4.

Lautenschlager U.; Eschenauer, H. A.; Mistree, F. 1996: Components of Turbo Systems—A Proposal for Finding Better Layouts. *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium* (held at Bellevue, WA). Paper 96-4096, pp. 1025–1035.

Lewis, R. M. 1996: A Trust Region Framework for Managing Approximation Models in Engineering Optimization. *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium* (held at Bellevue, WA). Paper 96-4101, pp. 1053–1055.

Pourazady, M.; Fu, Z. 1996: Integrated Approach to Structural Shape Optimization. *Computers and Structures* **60**, No. 2, 279–289.

Renaud, J. E.; Gabriele, G. A. 1994: Approximation in Non-Hierarchic System Optimization. *AIAA Journal* **32**, No. 1, 198–205.

Renaud, J. E.; Gabriele, G. A. 1993: Improved Coordination in Non-Hierarchic System Optimization. *AIAA Journal* **31**, No. 12, 2367–2373.

Renaud, J. E.; Gabriele, G. A. 1991: Sequential Global Approximation in Non-Hierarchic System Decomposition and Optimization. *Advances in Design Automation*, Design Automation and Design Optimization, *ASME Publication DE-Vol. 32-1*, G. Gabriel (Ed), pp. 191–200, 19th Design Automation Conference, Miami, FL, Sept. 22–25.

Rockafellar R. T. 1973: The Multiplier Method of Hestenes and Powell Applied to Convex Programming. *J. Optimization Theory and Applications* **12**, 555–562.

Rodríguez, J. F.; Renaud, J. E.; Watson, L. T. 1997: Trust Region Augmented Lagrangian Methods for Sequential Response Surface Approximation and Optimization. *Proceedings of the ASME Design Engineering Technical Conference* (held at Sacramento, CA). Paper DETC97/DAC3773.

Roux, W. J.; Stander, N.; Haftka, R. T. 1996: Response Surface Approximations for Structural Optimization. *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium* (held at Bellevue, WA). Paper 96-4042, 1053–1055.

Sellar, R. S. 1997: Multidisciplinary Design Using Artificial Neural Networks for Discipline Coordination and System Optimization. *Doctoral Dissertation*, Department of Aerospace and Mechanical Engineering, University of Notre Dame, April.

Sellar, R. S.; Batill, S. M. 1996c: Concurrent Subspace Optimization Using Gradient-Enhanced Neural Network Approximations. *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium* (held at Bellevue, WA). Paper 96-4019, pp. 319–330.

Sellar, R. S.; Stelmack, M. A.; Batill, S. M.; Renaud, J. E. 1996b: Multidisciplinary Design and Analysis of an Integrated Aeroelastic/Propulsive System. *37th*

AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference (held at Salt Lake City, UT). Paper 96-1383, 583–594.

Sellar, R. S.; Batill, S. M.; Renaud, J. E. 1996a: A Neural Network-Based, Concurrent Subspace Optimization Approach to Multidisciplinary Design Optimization. *34th AIAA Aerospace Sciences Meeting* (held at Reno, NV). Paper 96-1383.

Sellar, R. S.; Batill, S. M.; Renaud, J. E. 1994: Mixed Discrete/Continuous Optimization of Aircraft Systems Using Neural Networks. *Proceedings of the Fifth AIAA/USAF/NASA/ISSMO Symposium* (held at Panama City, FL). Paper 94-4348, pp. 910–921.

Shankar, J.; Ribbens, C. J.; Haftka, R. T.; Watson, L. T. 1993: Computational study of a nonhierarchical decomposition algorithm. *Comput. Optim. Appl.* **2**, 273–293.

Simpson T. W.; Chen, W.; Allen, J. K.; Mistree, F. 1996: Conceptual Design of a Family of Products Through the Use of the Robust Exploration Method. *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium* (held at Bellevue, WA). Paper 96-4161, pp. 1535–1545.

Sobieszczanski-Sobieski, J. 1990: Sensitivity of Complex, Internally Coupled Systems. *AIAA Journal* **28**, No. 1, January.

Sobieszczanski-Sobieski, J. 1988: Optimization by Decomposition: A Step from Hierarchic to Non-Hierarchic Systems. *Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization* (held at Hampton, VA). NASA Conference Publication 3031, Part 1. 28–30.

Swift, R. A.; Batill, S. M. 1993: Structural Design Space Definition Using Neural Networks and a Reduced Knowledge Base. *AIAA/AHS/ASEE Aerospace Design Conference* (held at Irvine, CA). Paper No. 93-1034-CP.

Swift, R. A.; Batill, S. M. 1992a: Simulated Annealing Utilizing Neural Networks For Discrete Variable Optimization Problems In Structural Design. *AIAA/-ASME/ASCE/AHS/ASC 33rd Structures, Structural Dynamics and Materials Conference* (held at Dallas, TX). Paper No. 92-2311-CP.

Swift, R. A.; Batill, S. M. 1992b: Damage Tolerant Structural Design Using Neural Networks. *AIAA Aerospace Design Conference and Exhibit* (held at Irvine, CA). Paper No. 92-1097-CP.

Swift, R. A.; Batill, S. M. 1991: Application of Neural Networks to Preliminary Structural Design. *Proceedings of the AIAA/ASME/ASCE/AHS/ASC 32nd Structures, Structural Dynamics and Materials Conference* (held at Baltimore, Maryland), pp. 335–343.

Thomas, H. L.; Vanderplaats, G. N.; Shyy, Y. K. 1992: A Study of Move Limit Adjustment Strategies in the Approximation Concepts Approach to Structural Synthesis. *Proceedings of the Fourth AIAA/USAF/-NASA/OAI Symposium on Multidisciplinary Analysis and Optimization* (held at Cleveland, OH), pp. 507–512.

Venter, G.; Haftka, R. T.; Starnes, J. H. 1996: Construction of Response Surfaces for Design Optimization Applications. *Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium* (held at Bellevue, WA). Paper 96-4040, pp. 548–564.

Wujek, B. A.; Renaud, J. E.; Batill, S. M. 1997: A Concurrent Engineering Approach for Multidisciplinary Design in a Distributed Computing Environment. In: N. Alexandrov and M.Y. Hussaini (eds), *Multidisciplinary Design Optimization: State-of-the-Art*, Proceedings in Applied Mathematics 80, pp. 189–208. Philadelphia: SIAM.

Wujek, B. A.; Renaud, J. E.; Batill, S. M.; Brockman, J. B. 1996: Concurrent Subspace Optimization Using Design Variable Sharing in a Distributed Computing Environment. *Concurrent Engineering: Research and Applications (CERA)*. Technomic Publishing Company, Lancaster, PA.