

ON MANAGING THE USE OF SURROGATES IN GENERAL NONLINEAR OPTIMIZATION AND MDO

NATALIA M. ALEXANDROV*

Abstract

This paper is concerned with a trust-region approximation management framework (AMF) for solving the nonlinear programming problem, in general, and multidisciplinary optimization problems, in particular. The intent of the AMF methodology is to facilitate the solution of optimization problems with high-fidelity models. While such models are designed to approximate the physical phenomena they describe to a high degree of accuracy, their use in a repetitive procedure, for example, iterations of an optimization or a search algorithm, make such use prohibitively expensive. An improvement in design with lower-fidelity, cheaper models, however, does not guarantee a corresponding improvement for the higher-fidelity problem. The AMF methodology proposed here is based on a class of multilevel methods for constrained optimization and is designed to manage the use of variable-fidelity approximations or models in a systematic way that assures convergence to critical points of the original, high-fidelity problem.

Key Words: Approximation concepts, model management, surrogate optimization, nonlinear programming, multidisciplinary optimization, multilevel methods, computational engineering.

Introduction

This paper concerns a globally convergent approach—a trust-region approximation management framework (AMF)—for controlling the use of approximations or models of variable fidelity in solving the nonlinear programming problem (NLP):

$$\begin{aligned} &\text{minimize} && f(x) \\ &\text{subject to} && h(x) = 0 \\ & && g(x) \geq 0, \end{aligned} \tag{1}$$

where the objective $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and the equality constraints $h : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and the inequality constraints $g : \mathbb{R}^n \rightarrow \mathbb{R}^p$ are sufficiently smooth nonlinear func-

tions with $m, n, p > 0$ and $m < n$.

Mathematical models of physical phenomena necessarily provide only an approximation to the “true” entities they describe, that is, all computational models are approximations or surrogates of the underlying function. In some publications, special meaning is reserved for the terms “approximation”, “model”, and “surrogate”, and the meaning differs from author to author. In the context of this paper, the terms are interchangeable.

High-fidelity models, such as the Navier-Stokes equations of aerodynamics, are designed to approximate physical phenomena to a high degree of accuracy. However, their use in repetitive procedures, for instance, in iterations of an optimization algorithm, can be prohibitively expensive. On the other hand, an improvement in design with lower-fidelity, cheaper models (e.g., the Euler equations for aerodynamics) does not guarantee a corresponding improvement for the higher-fidelity problem. A natural approach to alleviating this difficulty is to alternate the use of higher-fidelity approximations with the use of lower-fidelity approximations in a single optimization procedure, based on some rules for deciding when to switch to a model of a different fidelity.

Approximations have been used in engineering optimization for a long time in various procedures based on heuristics (e.g., [1, 2, 3]). A survey on the use of approximations in structural optimization can be found in [4]. Reports of recent efforts in developing methodologies for “variable-complexity modeling” can be found in [5, 6]. With a few exceptions ([7, 8]), until recently, the analysis had focused on the question of convergence to a solution of the surrogate problem ([9], [10]). The intent of the AMF methodology proposed here is to facilitate the solution of optimization problems with high-fidelity models by alternating their use with the use of lower-fidelity models in a systematic way, resulting in a procedure that is globally convergent to a critical point of the high-fidelity problem.

*Member AIAA. Research Scientist, MDOB, NASA Langley, Hampton, VA, 23681-2199, n.alexandrov@larc.nasa.gov. Copyright ©1998 by the American Institute of Aeronautics and Astronautics, Inc. No copyright is asserted in the United States under Title 17, U.S. Code. The U.S. Government has a royalty-free license to exercise all rights under the copyright claimed herein for Government Purposes. All other rights are reserved by the copyright owner.

Work on approximation management frameworks may be roughly categorized into zero-order [11] or derivative-free [12] methods and first-order methods [13]. Formally, the zero-order methods are those based on direct-search algorithms that do not build models of functions and, hence, require no derivative information from the user, while first-order methods explicitly rely on derivative-based models. In practice, however, the distinction is not as pronounced, because zero-order frameworks are hybrid methods that use a direct-search component but do build models with approximate derivative information. Because first-order frameworks rely on derivative-based models explicitly, however, to date only the first-order frameworks have been extended to handling general nonlinear constraints. The question of when it may be preferable to use a framework of a particular order is open and is a subject of ongoing research.

A provably convergent first-order AMF for unconstrained optimization was introduced in [13, 14, 15]. It was based on the trust-region methodology (e.g., [16, 17]), which can be described as an adaptive move limit strategy for improving the global behavior of local model-based algorithms. The work in [13] dealt with unconstrained optimization for two reasons. First, many algorithms for solving constrained problems are reduced to solving a sequence of unconstrained subproblems. Thus the analysis of unconstrained problems provides a foundation for the treatment of constrained ones. Second, the introduction of the idea was eased by addressing only unconstrained problems. However, the need to solve constrained problems was obvious, and in [13], we stated that the extension of the unconstrained AMF to constraints via, say, the augmented Lagrangian approach [18] was immediate. This extension is done in [19]. Another extension of the AMF in [13] to constrained optimization has been proposed in recent work [20] that uses multiplier methods together with response surfaces and the concurrent subspace optimization method [21] for multidisciplinary design optimization (MDO). Convergence of the latter method has been conjectured.

The AMF methodology discussed in the present paper concerns a scheme based on a class of multilevel methods for large-scale constrained trust-region optimization (MAESTRO) [22, 23, 24, 25]. The resulting AMF manages the use of variable-fidelity models both for the constraints and the objective in a systematic fashion that preserves the global convergence properties of the underlying class of algorithms.

The idea is as follows. Given high-fidelity models $f(x)$, $h(x)$, $g(x)$ of the objective, equality, and inequality constraints, respectively, of a physical pro-

cess, and a suite of corresponding lower fidelity models $\{a_i^f(x)\}$, $\{a_i^h(x)\}$, $\{a_i^g(x)\}$, the overall “external” framework is that of a suitably modified foundation algorithm, in this case, the MAESTRO class of algorithms. The computation of the trial steps in the external framework is then itself a set of iterative procedures. These procedures bear the brunt of the computational expense and they are done by using approximations to the lower-fidelity models. The algorithm resorts to high-fidelity computations only periodically to evaluate progress towards a critical point (or a solution) of the high-fidelity problem.

The multilevel AMF is capable of handling arbitrary models, provided that a set of consistency conditions defined later in the paper is satisfied at some points of the iterative procedure. For example, AMF is not limited to the use of algebraic, Taylor series, or response surface approximations. Analyses of various physical fidelity, such as Navier-Stokes and Euler codes, can be used as variable-fidelity function evaluators in AMF. In current demonstrations, we are particularly interested in managing aerodynamic models of varying physical fidelity.

The paper is organized as follows. Section 2 gives a brief overview of the underlying class of multilevel algorithms. Section 3 introduces the bilevel AMF intended for solving conventional NLP with a single block of constraints. Section 4 describes a multilevel AMF for large NLP or for MDO problems. Section 5 contains some very preliminary numerical results. Section 6 concludes with a summary and some remarks concerning ongoing research.

The notation will be introduced as necessary. All norms in the remainder of the paper are ℓ_2 norms.

Background: **MAESTRO class for nonlinear optimization**

This section contains a brief review of the underlying MAESTRO class of algorithms for constrained optimization [22, 23, 24, 25, 26] without the inclusion of the approximation management.

The current AMF demonstrations at the Multidisciplinary Optimization Branch (MDOB) deal with relatively small problems— $O(10)$ variables and constraints—with expensive function evaluations. The general inequalities are managed via squared slacks and, therefore, the remainder of the paper is concerned with the nonlinear equality constrained optimization problem (EQC):

$$\begin{aligned} &\text{minimize} && f(x) \\ &\text{subject to} && h(x) = 0. \end{aligned} \tag{2}$$

The MAESTRO class of methods for constrained

optimization is a class of algorithms motivated by MDO problems with arbitrary couplings but applicable to large-scale optimization problems in general.

In this approach, the constraints or equations of the system are partitioned into M blocks

$$h(x) = \begin{pmatrix} h_1(x) \\ \vdots \\ h_M(x) \end{pmatrix}$$

in a manner dictated by the application. In the context of MDO, disciplinary boundaries provide a natural partitioning into blocks. No special structure is imposed on the problem. In particular, narrow bandwidth of coupling is not assumed, and the subsystems of the problem can be, in principle, fully coupled.

Once the system is partitioned into subsystems, given a current approximation to a solution, the next point is computed in a sequence of progressively lower-dimensional trust-region subproblems, each of which computes substeps on its own block of equations or constraints, subject to maintaining the predicted improvement already obtained for the previous blocks. The final substep is computed by minimizing the model of the objective function or of the Lagrangian, subject to maintaining the predicted improvement in the models of all constraint blocks. The total step is the sum of the substeps computed in the subproblems.

In particular, suppose x_c is the current approximation to a solution of problem EQC. Let $\delta_1, \dots, \delta_{M+1}$ be the trust region radii for the M subproblems that deal with constraints plus the subproblem that deals with the objective function. The trust-region radii bound the regions in which particular models are “trusted” to approximate the behavior of the corresponding functions. (After the trial step is computed, the actual behavior of the merit function is compared to its predicted behavior and the trust-region radii are updated in a systematic manner.) Further, let $y_0 = x_c$. Then the trial step s_c is computed as a sum of the substeps s_k , each of which is an approximate solution of one of the $M+1$ subproblems. During the multilevel constraint elimination procedure, each substep s_k is computed by approximately minimizing the Gauss-Newton model of a particular constraint block, subject to the null space constraints that assure the maintenance of the predicted improvement already obtained for the previous blocks of constraints. Specifically, the following subproblem is solved:

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \left\| h_k(y_{k-1}) + \nabla h_k^T(y_{k-1}) s \right\|^2 \\ & \text{subject to} && \nabla h_j^T(y_{j-1}) s = 0, j = 1, \dots, k-1, \\ & && \|s\| \leq \delta_k, \end{aligned}$$

where $k = 1, \dots, M$ is the number of the currently processed constraint block and the number of the subproblem. (Note that for $k = 1$ there is no null-space constraint.) The points y_k are then set to $y_k + s_k$. Once the constraints have been processed, the substep s_{M+1} on the model of the objective function is computed by approximately minimizing the quadratic model $\phi(s)$ of the objective in its own trust region, as follows:

$$\begin{aligned} & \text{minimize} && \phi(s) \equiv f(y_M) + \nabla f^T(y_M) s + \frac{1}{2} s^T B_M s \\ & \text{subject to} && \nabla h_j^T(y_{j-1}) s = 0, j = 1, \dots, M \\ & && \|s\| \leq \delta_{M+1}, \end{aligned}$$

where B_M is the Hessian of f at y_M or an approximation to it. The only assumption placed on the approximation of the Hessian is that its norm is uniformly bounded from above. In particular, the Hessian can be zero, resulting in a linear model of the objective.

The total trial step s_c from the point x_c is the sum $s_c = s_1 + \dots + s_{M+1}$.

The salient feature of the trial step computation is that each substep s_k is assumed to solve its subproblem only approximately. “Approximately” means that each s_k can be obtained in any manner suitable to the application, as long as it satisfies two mild conditions. Specific ways of computing the substeps will give rise to different members of the MAESTRO class.

The first of the two conditions on the substeps is a sufficient decrease condition known as the Fraction of Cauchy Decrease condition (FCD). It concerns the change in the model of the objective function or constraints from the point x_c to the point $x_c + s_c$ for the entire trial step and from points y_{k-1} to points $y_{k-1} + s_k$ in computing the substeps. Roughly speaking, FCD says that a particular model is required to predict a fraction of the improvement in the objective or constraint functions that is predicted by the steepest descent step, i.e., the minimizer of the linear model of a particular function, restricted to the trust region. Let \hat{h}_k denote half the gradient of the norm squared of the constraints, i.e.,

$$\hat{h}_k \equiv \nabla h_k(y_{k-1}) h_k(y_{k-1}).$$

For each constraint block, the FCD condition can be stated in the following form: there exist $\sigma > 0$ and $\Gamma > 0$, independent of the iterates, for which the substep s_k satisfies

$$\begin{aligned} & \left\| h_k(y_{k-1}) \right\|^2 - \left\| h_k(y_{k-1}) + \nabla h_k(y_{k-1})^T s_k \right\|^2 \geq \\ & \sigma \left\| \hat{h}_k \right\| \min \left(\delta_k, \frac{\left\| \hat{h}_k \right\|}{\Gamma} \right). \end{aligned}$$

For the objective, the FCD condition implies that the substep s_{M+1} satisfies

$$f(y_M) - \phi(y_M + s_{M+1}) \geq \sigma \|\nabla f(y_M)\| \min\left(\delta_{M+1}, \frac{\|\nabla f(y_M)\|}{\Gamma}\right).$$

The constants σ and Γ need not be same for the objective and the constraint blocks, in general, but they can be made the same.

The FCD condition is easily satisfied in practice. Under the assumption of FCD, strong convergence statements can be made about trust-region algorithms for unconstrained optimization [17, 27]. For the MAESTRO class for constrained optimization, an additional boundedness condition is necessary. In particular, each substep s_k , $k = 1, \dots, M$ is required to satisfy

$$\|s_k\| \leq \mathcal{K} \|h_k(y_{k-1})\|$$

for some constant \mathcal{K} independent of the iterates. Again, \mathcal{K} need not be the same for all blocks of constraints. This condition is also very mild. It assures the existence of pseudoinverses for the derivative matrices of the constraint blocks, and it allows for a wide variety of step choices.

The trial step is evaluated by comparing the actual improvement in the merit function with the predicted improvement in the merit function. The step is accepted or rejected based on the comparison, and the trust-region radii are adjusted. The merit function and the step acceptance and update procedures are described in more detail in the next section.

Given the two conditions on the substeps s_k , under a number of conventional assumptions of the nonlinear programming analysis, the algorithms of the MAESTRO class have been proven to converge to critical points of problem EQC [22, 28].

The mildness of the requirements placed on the substeps assures that the class is very extensive. This attribute is significant from the application perspective, because in MDO, the blocks originate from different disciplines and almost certainly require different solution methods. Since each substep is computed autonomously, it has been noted ([22]) that MAESTRO would provide a natural framework for dealing with different models for MDO in a systematic fashion.

Bilevel AMF

We first consider the case of a single block of constraints, or the conventional NLP, in which the constraints are not partitioned. The trial step in the underlying algorithm is composed of two substeps, first

on the constraints and then on the objective, resulting in a bilevel MAESTRO-based AMF procedure.

In the MAESTRO class, as in classical trust-region algorithms, the amount of optimization done at each trial step is controlled by varying the size of the trust region, because only one model is used for constraints and one model is used for the objective during all optimization iterations. Because the multilevel AMF inherits the global convergence properties of the original class, it is not necessary to change models to obtain convergence. However, in the case when other models are available, such as varieties of structural or aerodynamic models, MAESTRO-based AMF provides the user with guidance on changing the nature of the models.

Computing the trial step

In the bilevel AMF, to compute the trial step from the current point x_c , we first select a model of the constraints a_c^h that satisfies the following consistency conditions for the constraints at that point:

$$\begin{aligned} a_c^h(x_c) &= h(x_c) \\ \nabla a_c^h(x_c) &= \nabla h(x_c). \end{aligned} \quad (3)$$

We find a substep s_1 that approximately minimizes that model within its trust region. The process of computing the substep is itself an iterative procedure. Next, we select a model a_c^f of the objective function or the Lagrangian that satisfies the consistency conditions at the just computed point:

$$\begin{aligned} a_c^f(x_c + s_1) &= f(x_c + s_1) \\ \nabla a_c^f(x_c + s_1) &= \nabla f(x_c + s_1). \end{aligned} \quad (4)$$

The substep s_2 is computed in another iterative procedure to approximately minimize the model in its own trust region. The total trial step s_c is the sum of the two substeps.

Consistency conditions (3) and (4) are the conditions used in the unconstrained AMF [13]. The conditions can be relaxed, but we will not pursue that line of reasoning here. Even in the form of (3) and (4), the conditions are not restrictive, as they can be easily enforced [29], given any two models.

Consistency requirements are placed on the models used at each optimization iteration of the external framework. No such assumptions are made during the inner iterative procedures when the approximations of the lower-fidelity models are minimized repeatedly.

Numerically, let $\Delta_c \in \Re$ be the current value of the trust-region radius. Then the current trial step s_c will be computed as follows:

Set $z_0 = x_c$
 Select a_c^h , such that $a_c^h(z_0) = h(z_0)$ and $\nabla a_c^h(z_0) = \nabla h(z_0)$.
 Compute s_1 that approximately solves:
 minimize $a_c^h(z_0 + s)$
 subject to $\|s\| \leq \theta \Delta_c$, $\theta \in (0.5, 0.6)$.
 Set $z_1 = z_0 + s_1$.
 Select a_c^f , such that $a_c^f(z_1) = f(z_1)$ and $\nabla a_c^f(z_1) = \nabla f(z_1)$.
 Compute s_2 that approximately solves:
 minimize $a_c^f(z_1 + s)$
 subject to $\nabla a_c^h(z_0)^T s = 0$
 $\|s\| \leq \sqrt{\Delta_c^2 - \|s_1\|^2}$.
 Set $s_c = s_1 + s_2$,

where $\theta \in (0.5, 0.6)$ is a somewhat arbitrary constant (between zero and one) that partitions the trust region into two sections, one for work with the constraints, the other, for work with the objective.

This algorithm for computing the trial step is a special case of the MAESTRO class with the distinction that the Gauss-Newton model of the constraints and the quadratic model of the objective or the Lagrangian have been replaced by the general, first-order models that satisfy the consistency conditions (3) and (4).

For the purposes of global convergence analysis, the ℓ_2 penalty function

$$\mathcal{P}(x; \rho) \equiv f(x) + \rho \|h(x)\|^2$$

suffices as a merit function. The penalty parameter ρ balances the progress toward optimality with the progress toward feasibility. Global convergence analysis is also applicable when the augmented Lagrangian is used as the merit function. The latter would be used for computational efficiency.

Updating the penalty parameters

Before the trial step is evaluated, the penalty parameter is updated. The parameter ρ is a positive, scale dependent number. The updating scheme is rigorous and it extends the scheme introduced in [30]. The penalty parameter is occasionally increased to ensure that the predicted reduction in the merit function is positive at each iteration. In particular, let us set $\rho_c = 1$ at the beginning and choose $\beta \in (0, 1)$. Let $s_c = s_1 + s_2$ be the trial step computed during some iteration, and let ρ_c be the current value of the penalty parameter. Then we update ρ_c as follows:

Compute $hpred_c \equiv \|h(x_c)\|^2 - \|a_c^h(x_c + s_c)\|^2$.
 Compute the predicted reduction
 $pred_c \equiv [f(x_c) - a_c^f(x_c + s_c)] + \rho_c hpred_c$.

If $pred_c \geq \frac{\rho_c}{2} hpred_c$ then

$$\rho_+ = \rho_c.$$

Else

$$\rho_+ = \rho_c^- + \beta,$$

$$\text{where } \rho_c^- = \frac{2[a_c^f(x_c + s_c) - f(x_c)]}{hpred_c}.$$

End if

Set $\rho_c = \rho_+$.

Thus, the penalty parameter is increased just sufficiently to maintain positive predicted reduction. Given that the algorithm does not terminate at a particular iterate, the penalty parameter can be shown to be bounded above. Since, when the penalty parameter has to be increased, it is increased just enough to keep the predicted reduction positive, it is expected to grow more slowly than in methods where it is increased simply by a predetermined factor. It should be emphasized that the penalty parameter is used only to evaluate the trial step and not in computing the step.

Evaluating the trial step and updating the radius

Once the penalty parameter is updated, the trial step is evaluated. It is accepted if it produces an actual decrease in the merit function, and it is rejected otherwise:

$$x_+ = \begin{cases} x_c + s_c & \text{if } \mathcal{P}(x_c + s_c; \rho_c) < \mathcal{P}(x_c; \rho_c) \\ x_c & \text{otherwise.} \end{cases} \quad (5)$$

To adjust the trust-region radius, one considers the ratio of the actual reduction defined as

$$ared_c = \mathcal{P}(x_c; \rho_c) - \mathcal{P}(x_c + s_c; \rho_c),$$

to the predicted reduction $pred_c$:

$$r = \frac{ared_c}{pred_c}.$$

When the ratio is close to zero, the predictive behavior of the model is unsatisfactory. The agreement between the actual and predicted reduction need not be large. For instance, typically, $r < 10^{-5}$ is usually considered to signify a unsuccessful step. A larger ratio indicates satisfactory behavior. The radius is updated as follows: positive constants $r_1 < r_2 < 1$ and $c_1 < 1$, $c_2 > 1$ are chosen to regulate the contraction and expansion of the trust region. Then

$$\Delta_+ = \begin{cases} \max\{c_1 \|s_c\|, \Delta_*\} & \text{if } r < r_1 \\ \min\{c_2 \|\Delta_c\|, \Delta^*\} & \text{if } r > r_2 \\ \|s_c\| & \text{otherwise,} \end{cases} \quad (6)$$

where Δ_* is a lower bound on the trust-region radius and Δ^* is an upper bound. Details on typical values

of the parameters for trust-region algorithms can be found in [31].

One should note that although for the purposes of engineering analysis a large r (greater than one) indicates a poor match between the higher and lower fidelity models, for the purposes of optimization, a large r indicates excellent predictive capabilities of the model, and the radius Δ_c is increased.

Details of computing the trial step

Since the underlying algorithm belongs to the MAESTRO class, the MAESTRO-based AMF will converge to a critical point of the high-fidelity problem under the assumptions that lead to convergence of the underlying class. This means finding the substeps s_1 and s_2 that will satisfy the sufficient decrease conditions necessary for establishing convergence. The following procedure will suffice. (We will use v_j and p_j as local variables in both procedures.)

Compute the substep s_1 on the constraints in an iterative procedure of minimizing Gauss-Newton models of a_c^h within the trust region of that subproblem, using $\|a_c^h\|^2$ as an internal merit function:

Given $z_0 = x_c \in \mathbb{R}^n$, $\Delta_c > 0$, $\tau \in (0, 1)$,
and $\alpha \geq 1$, set $y_0 = z_0$, $\delta_0 = \tau\Delta_c$, $v_0 = 0$.
For $j = 0, 1, \dots$, while $\|v_j\| \leq \alpha\Delta_c$
and at least until $v_j \neq 0$ do {

Construct

$$q_j^h(y_j + p) \equiv \|a_c^h(y_j) + \nabla a_c^h(y_j)^T p\|^2.$$

Find an approximate solution p_j to

$$\begin{aligned} &\text{minimize} && q_j^h(y_j + p) \\ &\text{subject to} && \|p\| \leq \delta_j \\ &&& \|y_j + p\| \leq \Delta_c \end{aligned}$$

that satisfies FCD for $\|a_c^h\|^2$ from y_j .

Compare the actual and predicted decrease in $\|a_c^h\|^2$:

$$r = \frac{\|a_c^h(y_j)\|^2 - \|a_c^h(y_j + p_j)\|^2}{\|a_c^h(y_j)\|^2 - q_j^h(y_j + p_j)}.$$

Evaluate p_j and update y_j according to (5) and δ_j according to (7).

Set $v_{j+1} = v_j + (y_{j+1} - y_j)$.

}

Set $s_1 = v_j$.

Thus, the subproblem for computing s_1 is an unconstrained trust-region subproblem. By results of the unconstrained AMF [13], the substep s_1 satisfies the FCD condition for that subproblem.

Once s_1 is computed, z_1 is set to $z_0 + s_1$, and s_2 can be computed via the following procedure:

Given $z_1 \in \mathbb{R}^n$, $\Delta_c > 0$, $\tau \in (0, 1)$, and $\alpha \geq 1$,

set $y_0 = z_1$, $\delta_0 = \tau\sqrt{\Delta_c^2 - \|s_1\|^2}$, $v_0 = 0$.

For $j = 0, 1, \dots$, while $\|v_j\| \leq \alpha\sqrt{\Delta_c^2 - \|s_1\|^2}$
and at least until $v_j \neq 0$ do {

Construct

$$q_j^f(y_j + p) \equiv a_c^f(y_j) + \nabla a_c^f(y_j)^T p + \frac{1}{2} p^T B_j p,$$

where B_j approximates the second order information for a_c^f at y_j .

Find an approximate solution p_j to

$$\begin{aligned} &\text{minimize} && q_j^f(y_j + p) \\ &\text{subject to} && \nabla c_k^h(z_0)^T p = 0 \\ &&& \|p\| \leq \delta_j \\ &&& \|y_j + p\| \leq \Delta_c \end{aligned}$$

that satisfies FCD for a_c^f from y_j .

Compare the actual and predicted decrease in a_c^f :

$$r = \frac{a_c^f(y_j) - a_c^f(y_j + p_j)}{a_c^f(y_j) - q_j^f(y_j + p_j)}.$$

Evaluate p_j and update y_j according to (5) and δ_j according to (7).

Set $v_{j+1} = v_j + (y_{j+1} - y_j)$.

}

Set $s_2 = v_j$.

Set $s_c = s_1 + s_2$.

To prove the satisfaction of FCD for the resulting step s_c , a more stringent rule for updating y_j is required than for updating x_c in (5). Namely, $\mu > 0$, independent of the iterates, is chosen and the iterate is updated as follows:

$$\begin{aligned} \text{If } y_j = x_c, \text{ then } y_{j+1} &= \begin{cases} y_j + p_j & \text{if } r > \mu \\ y_j & \text{otherwise.} \end{cases} \\ \text{If } y_j \neq x_c, \text{ then } y_{j+1} &= \begin{cases} y_j + p_j & \text{if } r > 0 \\ y_j & \text{otherwise.} \end{cases} \end{aligned} \quad (7)$$

Note that for computing s_2 , the null-space constraints that preserve the predicted decrease in the constraints already obtained during the computation of s_1 , are eliminated via a change of variables, thus converting the second subproblem into a lower-dimensional unconstrained trust-region subproblem. By results for the unconstrained AMF, s_2 satisfies FCD for its subproblem.

Note, again, that in the ‘‘inner’’ algorithms, the models that are minimized are now the quadratic approximations of a_c^h and a_c^f , not of h and f , respectively.

There are a number of ways to solve the subproblems in the ‘‘inner’’ iterative procedures. The reader is referred to [24, 28] for details.

Statement of the bilevel AMF algorithm

The algorithm in its entirety can now be stated concisely:

Given $x_c \in \mathbb{R}^n$, $\Delta_c > 0$, and $\rho_c \geq 1$,

Do until convergence:

Select the models a_c^h and a_c^f .
 Compute the trial step $s_c = s_1 + s_2$.
 Update the penalty parameter ρ_c .
 Evaluate s_c and update x_c and Δ_c .

End Do

Multilevel AMF

In the context of MDO, the bilevel algorithm is easily extended to the general, multilevel case. In particular, the constraint system is partitioned into M blocks, $h_1(x), \dots, h_M(x)$, and the total trial step is computed as a sum of $M + 1$ substeps.

We describe a multilevel algorithm with general first-order models. We will denote the iterates x_c and will omit the subscripts denoting the iteration number, for convenience of notation. Instead, the subscripts will denote entities concerning a particular constraint block, e.g., we let a_1^h, \dots, a_M^h denote the models of the constraint blocks $h_1(x), \dots, h_M(x)$, respectively. Then the algorithm can be stated as follows:

Given $x_c \in \mathbb{R}^n$, $\delta_k > 0, k = 1, \dots, M + 1$, $\rho_k > 0, k = 1, \dots, M$,

Outer Loop: Do until convergence:

$z_0 = x_c$

Compute the trial step

Inner Loop: Do $k = 1, M$

Select a_k^h such that $a_k^h(z_{k-1}) = h_k(z_{k-1})$
 and $\nabla a_k^h(z_{k-1}) = \nabla h_k(z_{k-1})$

Compute s_k that approximately solves:

minimize $a_k^h(z_{k-1} + s)$
 subject to $\nabla a_j^h(z_{j-1})^T s = 0, j = 1, \dots, k - 1$
 $\|s\| \leq \delta_k$

Set $z_k = z_{k-1} + s_k$

End Inner Loop

Step on the objective function:

Select a^f such that $a^f(z_M) = f(z_M)$
 and $\nabla a^f(z_M) = \nabla f(z_M)$

Compute s_{M+1} that approximately solves:

minimize $a^f(z_M + s)$
 subject to $\nabla a_j^h(z_{j-1})^T s = 0, j = 1, \dots, M$
 $\|s\| \leq \delta_{M+1}$

The trial step is: $s_c = s_1 + \dots + s_{M+1}$

Update the penalty parameters

Evaluate the step and update

the trust-region radii

End Outer Loop

In summary, at each iteration, for each of the subproblems, a lower-fidelity approximation is selected

that satisfies the consistency conditions for that particular model at the most recently computed point. Then unconstrained procedures described in the section on bilevel AMF are applied to models of each constraint block, followed by the unconstrained procedure on the model of the objective function. The procedures are unconstrained because the null-space constraints on all of the subproblems are eliminated via a change of variables. The individual substeps $s_k, k = 1, \dots, M + 1$, satisfy FCD for the subproblems they solve.

The individual merit functions are modified in an obvious manner. The complete merit function of the multilevel algorithm has to be used with the appropriate “staged” penalty parameter updating procedure. The reader is referred to [25], for example, for details on the total merit function and procedures for updating the penalty parameters and the trust-region radii.

Preliminary numerical results

The algorithm has undergone initial testing on several simple problems. Table 1 illustrates preliminary results. Problems denoted by “HSxx” are the problems from the Hock and Schittkowski test suite for nonlinear optimization [32, 33]. All problems are small, algebraic, nonlinear programming problems with well-known solutions.

The entries in the tables are the numbers of function evaluations taken by each method to attain a known solution of a particular problem.

The problems were initially solved using the well-known commercial optimization package NPSOL [34] that improves the global properties of the SQP algorithm via the use of line searches. The problems were then solved using a research implementation of MAESTRO without approximation management. For NPSOL and MAESTRO, the number of function evaluations is reported. “F” denotes failure to converge in a thousand iterations.

Finally, the problems were solved with the MAESTRO-based AMF approach. Because the present demonstrations deal with physical models of varying fidelity, this very simple test used function evaluations computed to machine precision as the “high-fidelity” models, and function evaluations with abbreviated precision and added noise as “lower-fidelity” models. A variety of precisions were attempted. On average, the “lower-fidelity” functions have from two to four digits of accuracy after the decimal point. The number of “high-fidelity” evaluations is reported with the total number of evaluations, including the number “lower-fidelity” evalua-

tions, given in parentheses.

Given the initial problem set, the results suggest that the MAESTRO-based AMF tends not impact the performance of the underlying algorithm significantly and it does reduce the number of “high-fidelity” function evaluations.

In order to draw definitive conclusions, a considerable amount of experimentation with physics-based models will be required.

At present, we are continuing testing the method on a subset of the MDO Test Suite [35] and we are in the process of demonstrating AMF on a more realistic problem of aerodynamic wing design.

Concluding remarks

In summary, we introduced an approximation management framework for solving constrained nonlinear problems based on an extension of the MAESTRO algorithms for nonlinear programming and MDO.

Global convergence of the AMF to a critical point of the original high-fidelity problem is an immediate consequence of the convergence results for underlying MAESTRO algorithms and the results for the unconstrained AMF.

The practical performance of any model-based algorithm will depend on the predictive qualities of the surrogates. We emphasize that even though the surrogate models may not be good approximators of the higher-fidelity models for the purposes of analysis, we are interested in predictive properties of the models strictly for the purposes of optimization. That is, an approximation may not capture all the important properties of a higher-fidelity function, but it may still produce a step that will lead to a satisfactory decrease in the higher-fidelity model.

For problems with a large number of general inequality constraints, using squared slacks may prove unsatisfactory. We are considering incorporation of AMF into the MAESTRO-based algorithm for handling problems with general inequalities explicitly [26].

Computational testing is in progress on a subset of the MDO Test Suite and a realistic problem of an aerodynamic wing design. Current work includes improvements in the implementation and research on various modeling alternatives and other underlying algorithms.

Acknowledgments

The author would like to thank Phil Gill for making the NPSOL package available to her.

References

- [1] L. A. Schmit, Jr. and B. Farshi. Some approximation concepts for structural synthesis. *AIAA Journal*, 12(5):692–699, 1974.
- [2] L. A. Schmit, Jr. and C. Fleury. Structural synthesis by combining approximation concepts and dual methods. *AIAA Journal*, 18:1252–1260, 1980.
- [3] L. A. Schmit, Jr. and H. Miura. Approximation concepts for efficient structural synthesis. Technical Report CR-2552, NASA, 1976.
- [4] J.-F. M. Barthelemy and R. T. Haftka. Approximation concepts for optimum structural design—a review. *Structural Optimization*, 5:129–144, 1993.
- [5] S. Burgee, A.A. Giunta, V. Balabanov, B. Grossman, W.H. Mason, R. Narducci, R.T. Haftka, and L.T. Watson. A coarse-grained parallel variable-complexity multidisciplinary optimization problem. *The International Journal of Supercomputing Applications and High Performance Computing*, 10(4):269–299, Winter 1996.
- [6] A.A. Giunta. *Aircraft Multidisciplinary Optimization Using Design of Experiments Theory and Response Surface Modeling Methods*. PhD thesis, Department of Aerospace and Ocean Engineering, Virginia Tech, Blacksburg, VA, 1997.
- [7] M. Avriel and A. C. Williams. Complementary geometric programming. *SIAM Journal on Applied Mathematics*, 19(1):125–141, July 1970.
- [8] A. J. Morris. Approximation and complementary geometric programming. *SIAM Journal on Applied Mathematics*, 23(4):527–531, December 1972.
- [9] V. Braibant and C. Fleury. An approximation-concepts approach to shape optimal design. *Computer Methods in Applied Mechanics and Engineering*, 53:119–148, 1985.
- [10] P. Hajela. Geometric programming strategies in large-scale structural synthesis. *AIAA Journal*, 24(7):1173–1178, 1986.
- [11] A.J. Booker, J. E. Dennis Jr., P.D. Frank, D.B. Serafini, V.Torczon, and M.W. Trosset. A rigorous framework for optimization of expensive

Problem	Starting Point	# Fn Evaluations		
		NPSOL	MAESTRO	MAESTRO AMF
HS6	-1.2 1	20	11	6(14)
	12 10	11	11	5(11)
	-10 0	30	13	6(17)
HS7	2 2	17	11	6(11)
	-35 -40	41	33	27(40)
	-15 -6	59	17	10(11)
HS26	0 0 0	49	21	15(25)
	5 -5 5	105	32	23(68)
	30 35 40	70	21	17(21)
HS39	2 2 2 2	16	19	8(24)
	40 2 4 -5	320	121	61(133)
	-2 -4 6 2	233	19	21(38)
HS40	-1 -1 -1 -1	135	11	10(23)
	0 -0.5 1 0	F	12	8(12)
	30 29 -39 3	40	61	44(83)
HS60	2 2 2	12	11	9(14)
	-10 40 9	124	21	11(36)
	100 100 -100	F	21	16(40)
HS77	2 2 2 2 2	21	11	6(17)
	10 10 10 10 10	F	28	16(34)
	20 20 20 20 20	F	27	15(56)

Table 1: Preliminary numerical results.

functions by surrogates. *Structural Optimization*, To appear.

- [12] A. R. Conn and Ph. L. Toint. An algorithm using quadratic interpolation for unconstrained derivative free optimization. In G. Di Pillo and F. Giannessi, editors, *Nonlinear Optimization and Applications*. Plenum Press, New York, 1996. Proceedings of the International School of Mathematics "G. Stampacchia" 21st Workshop, Erice, Italy, June 13-21, 1995.
- [13] N.M. Alexandrov, J. E. Dennis, Jr., R.M. Lewis, and V. Torczon. A trust region framework for managing the use of approximation models in optimization. *Structural Optimization*, 15(1):16-23, 1998.
- [14] N.M. Alexandrov. A trust region framework for managing approximation models in engineering optimization. In *Proceedings of the Sixth AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. AIAA, 1996. AIAA paper AIAA-96-4102.
- [15] R.M. Lewis. A trust region framework for managing approximation models in engineering optimization. In *Proceedings of the Sixth AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Design*, September 1996. AIAA paper 96-4101.
- [16] J. E. Dennis, Jr. and R.B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, Englewood Cliffs, NJ, 1983.
- [17] J.J. Moré. Recent developments in algorithms and software for trust region methods. In A. Bachem, M. Grötschel, and B. Korte, editors, *Mathematical Programming. The State of the Art, Bonn 1982*, pages 258-287. Springer-Verlag, Berlin, 1983.
- [18] A.R. Conn, N.I.M. Gould, and P.L. Toint. A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds. *SIAM Journal on Numerical Analysis*, 28(2):545-572, April 1991.
- [19] R.M. Lewis. Using sensitivity information in the construction of surrogates. In *Proceedings of the ISSMO/NASA First Internet Conference on Approximations and Fast Reanalysis in Engineering Optimization*, June 1998.

- [20] J. Rodríguez, J.E. Renaud, and L.T. Watson. Trust region augmented Lagrangian methods for sequential response surface approximation and optimization. In *Proceedings of DETC '97*, September 1997. ASME paper DETC97DAC3773, presented at the 1997 ASME Design Engineering Technical Conferences, September 14–17, Sacramento, California.
- [21] J.E. Renaud. A concurrent engineering approach for multidisciplinary design in a distributed computing environment. In N.M. Alexandrov and M.Y. Hussaini, editors, *Multidisciplinary Design Optimization: State of the Art*. SIAM, 1997.
- [22] N.M. Alexandrov. *Multilevel algorithms for nonlinear equations and equality constrained optimization*. PhD thesis, Rice University, Department of Computational and Applied Mathematics, May 1993. Also available as Technical Report TR93–20.
- [23] N.M. Alexandrov. Multilevel and multiobjective optimization in multidisciplinary design. In *Proceedings of the Sixth AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. AIAA, 1996. AIAA paper AIAA-96-4122.
- [24] N.M. Alexandrov. Multilevel methods for MDO. In N.M. Alexandrov and M. Y. Hussaini, editors, *Multidisciplinary Design Optimization: State of the Art*, pages 79–89. Society for Industrial and Applied Mathematics, Philadelphia, 1997.
- [25] N.M. Alexandrov and J. E Dennis, Jr. Multilevel algorithms for nonlinear optimization. In J. Borggaard, J. Burkardt, M. Gunzburger, and J. Peterson, editors, *Optimal Design and Control*, pages 1–22. Birkhäuser, 1994.
- [26] N.M. Alexandrov and M. El-Alem. A trust region algorithm for the general nonlinear programming problem. In preparation.
- [27] M.J.D. Powell. Convergence properties of a class of minimization algorithms. In O.L. Mangasarian, R.R. Meyer, and S.M. Robinson, editors, *Nonlinear Programming 2*, pages 1–27. Academic Press, New York, NY, 1975.
- [28] N.M. Alexandrov and J. E Dennis, Jr. A class of general trust-region multilevel algorithms for nonlinear optimization: Global convergence analysis. *SIAM Journal on Optimization*, 1998. submitted.
- [29] K.J. Chang, R.T. Haftka, G.L. Giles, and Pi-Jen Kao. Sensitivity-based scaling for approximating structural response. *Journal of Aircraft*, 30(2):283–288, March–April 1993.
- [30] M. El-Alem. *A Global Convergence Theory for a Class of Trust Region Algorithms for Constrained Optimization*. PhD thesis, Rice University, Department of Mathematical Sciences, 1988.
- [31] D.M. Gay. On convergence testing in model/trust-region algorithms for unconstrained optimization. Technical Report Computing Science Technical Report No. 104, Bell Laboratories, September 1982.
- [32] W. Hock and K. Schittkowski. Test examples for nonlinear programming codes. In *Lecture Notes in Economics and Mathematical Systems #187*. Springer-Verlag, Berlin, Heidelberg and New York, 1981.
- [33] W. Hock and K. Schittkowski. More test examples for nonlinear programming codes. In *Lecture Notes in Economics and Mathematical Systems #282*. Springer-Verlag, Berlin, Heidelberg and New York, 1987.
- [34] P.E. Gill, G.H. Golub, W. Murray, and M.H. Wright. *User's guide for SOL/NPSOL: a Fortran package for nonlinear programming*. Department of Operations Research, Stanford University, Stanford, CA, 1983.
- [35] S.L. Padula, N.M. Alexandrov, and L.L. Green. MDO Test Suite at NASA Langley Research Center. In *Proceedings of the Sixth AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Bellevue, WA. AIAA, 1996.