



**QUEEN'S
UNIVERSITY
BELFAST**

Surrogate-Based Optimization Using Multifidelity Models with Variable Parameterization and Corrected Space Mapping

Robinson, T., Eldred, M. S., Willcox, K. E., & Haines, R. (2008). Surrogate-Based Optimization Using Multifidelity Models with Variable Parameterization and Corrected Space Mapping. *AIAA Journal*, 46(11), 2814-2822. <https://doi.org/10.2514/1.36043>

Published in:
AIAA Journal

Queen's University Belfast - Research Portal:

[Link to publication record in Queen's University Belfast Research Portal](#)

General rights

Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The Research Portal is Queen's institutional repository that provides access to Queen's research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact openaccess@qub.ac.uk.

Surrogate-Based Optimization Using Multifidelity Models with Variable Parameterization and Corrected Space Mapping

T. D. Robinson*

Queen's University of Belfast, Belfast, Northern Ireland BT7 INN, United Kingdom

M. S. Eldred[†]

Sandia National Laboratories, Albuquerque, New Mexico 87185

and

K. E. Willcox[‡] and R. Haimes[§]

Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

DOI: 10.2514/1.36043

Surrogate-based-optimization methods provide a means to achieve high-fidelity design optimization at reduced computational cost by using a high-fidelity model in combination with lower-fidelity models that are less expensive to evaluate. This paper presents a provably convergent trust-region model-management methodology for variable-parameterization design models: that is, models for which the design parameters are defined over different spaces. Corrected space mapping is introduced as a method to map between the variable-parameterization design spaces. It is then used with a sequential-quadratic-programming-like trust-region method for two aerospace-related design optimization problems. Results for a wing design problem and a flapping-flight problem show that the method outperforms direct optimization in the high-fidelity space. On the wing design problem, the new method achieves 76% savings in high-fidelity function calls. On a bat-flight design problem, it achieves approximately 45% time savings, although it converges to a different local minimum than did the benchmark.

Introduction

AS COMPUTATIONAL capabilities continue to grow, designers of engineering systems have available an increasing range of numerical analysis models. These models range from low-fidelity simple-physics models to high-fidelity detailed computational simulation models. The drive toward including higher-fidelity analyses in the design process (for example, through the use of computational fluid dynamic analyses) leads to an increase in computational expense. As a result, design optimization, which requires large numbers of analyses of objectives and constraints, becomes prohibitively expensive for many systems of interest. This paper presents a methodology for improving the computational efficiency of a high-fidelity design. This method exploits variable fidelity and variable parameterization (that is, inexpensive models of lower physical resolution combined with coarser design descriptions) in a design optimization framework.

Surrogate-based optimization (SBO) methods have previously been proposed to achieve high-fidelity design optimization at reduced computational cost. In SBO, a surrogate, or less expensive and lower-fidelity model, is used for the majority of the optimization, with recourse to the high-fidelity analysis less frequently. The surrogate can be developed in a number of ways: for example, by using a simplified-physics model with a different set of governing equations. However, an improvement in a design predicted by a low-

fidelity model does not guarantee an improvement in the high-fidelity problem.

Past work has focused on providing surrogates that are computationally efficient to evaluate. These models can be roughly divided into three categories: data-fit surrogates such as response surfaces [1,2]; kriging [3]; radial basis functions [4] or extended radial basis functions [5,6]; reduced-order models, derived using techniques such as proper orthogonal decomposition [7] and modal analysis [8]; and hierarchical models, also called multifidelity, variable-fidelity, or variable-complexity models. In the latter case, a physics-based model of lower-fidelity and reduced computational cost is used as the surrogate in place of the high-fidelity model. The multifidelity case can be further divided based on the means by which the fidelity is reduced in the lower-fidelity model. The low-fidelity model can be the same as the high-fidelity, but converged to a higher residual tolerance [9]; it can be the same model on a coarser grid [10]; or it can use a simpler engineering model that neglects some physics modeled by the high-fidelity method [11]. Jones [12] compared a number of surrogates for use in global optimization.

Much work has been performed on developing SBO methods that are provably convergent to an optimum of the high-fidelity problem. Queipo et al. [13] reviewed a broad spectrum of SBO work. One promising group of methods is based on trust-region model management (TRMM), which imposes limits on the amount of optimization performed using the low-fidelity model, based on a quantitative assessment of that model's predictive capability. TRMM evolved from classical trust-region algorithms [14], which use quadratic surrogates, and has more recently been used for surrogates of any type [15]. These TRMM methods are provably convergent to an optimum of the high-fidelity model [16,17], provided the low-fidelity model is corrected to be at least first-order consistent with the high-fidelity model. Correcting to second-order or quasi-second-order consistency provides improved performance [18]. Yuan [19] presented a survey of unconstrained trust-region methods.

A number of researchers have developed SBO methods for constrained problems. Booker et al. [20] developed a direct-search SBO framework that converges to a minimum of an expensive objective function subject only to bounds on the design variables and that does not require derivative evaluations. Audet et al. [21]

Received 7 December 2007; revision received 4 April 2008; accepted for publication 5 May 2008. Copyright © 2008 by T. D. Robinson, M. S. Eldred, K. E. Willcox, and R. Haimes. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. Copies of this paper may be made for personal or internal use, on condition that the copier pay the \$10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0001-1452/08 \$10.00 in correspondence with the CCC.

*Lecturer in Aerospace Engineering; t.d.robinson@qub.ac.uk. Member AIAA.

[†]Principal Member of the Technical Staff, Optimization and Uncertainty Estimation Department. Associate Fellow AIAA.

[‡]Associate Professor of Aeronautics and Astronautics, Aerospace Computational Design Laboratory. Senior Member AIAA.

[§]Principal Research Engineer, Aerospace Computational Design Laboratory. Member AIAA.

extended that framework to handle general nonlinear constraints using a filter method for step acceptance [22]. Rodriguez et al. [23] developed a gradient-based TRMM augmented-Lagrangian strategy using response surfaces and showed that using separate response surfaces for the objective and constraints provided faster convergence than using a single response surface for the augmented Lagrangian. Alexandrov et al. [10] developed the MAESTRO class of methods, which use gradient-based optimization and trust-region model management, and compared them to a sequential quadratic programming (SQP)-like TRMM method. Under fairly mild conditions on the models, these methods are also convergent to a local minimum of the constrained high-fidelity problem [16,24]. Sadjadi and Ponnambalam [25] reviewed a broad spectrum of trust-region methods for constrained optimization, and Conn et al. [16] gave an extensive bibliography relating to both classical trust-region methods and more recent TRMM methods, for both the unconstrained and the constrained cases.

The SBO methods developed to date achieve computational gain by performing most of the analysis on the low-fidelity model; however, they require that the high- and low-fidelity models operate with the same set of design variables. For practical design applications, however, multifidelity models are often defined over different design spaces.

New methodology is therefore required for expanding surrogate-based design optimization to the case in which the low- and high-fidelity models use different design variables. Further, combining a low-fidelity model with a coarser parameterization of the design offers the opportunity for additional reduction in computational complexity and cost beyond current SBO methods. To achieve this, new design methodology is required that incorporates variable-parameterization models into SBO methods.

We consider a general design problem posed using the following nonlinear optimization formulation:

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to } \mathbf{c}(\mathbf{x}) \leq 0 \quad (1)$$

where $f: \mathbb{R}^n \rightarrow \mathbb{R}$ represents the scalar objective to be minimized, and $\mathbf{x} \in \mathbb{R}^n$ is the vector of n design variables that describe the design. The vector function $\mathbf{c}: \mathbb{R}^n \rightarrow \mathbb{R}^m$ contains m constraints, which provide a mathematical description of requirements that the design must satisfy. Both f and \mathbf{c} are assumed to be continuous and differentiable over the design space of interest. For realistic design problems of engineering relevance, the complexity of the optimization problem (1) is twofold: first, the simulations required to evaluate $f(\mathbf{x})$ and $\mathbf{c}(\mathbf{x})$ may be computationally expensive, and second, the dimensionality of \mathbf{x} may be large.

It is assumed in this discussion that a lower-fidelity model is available. This model is both less accurate and less computationally expensive. The lower-fidelity model for $f(\mathbf{x})$ is denoted as $\hat{f}(\hat{\mathbf{x}})$ and that for $\mathbf{c}(\mathbf{x})$ is $\hat{\mathbf{c}}(\hat{\mathbf{x}})$. The dimension of $\hat{\mathbf{x}}$, denoted as \hat{n} , may be different from the dimension of \mathbf{x} , denoted as n .

Some terminology is required as part of this discussion. A variable-fidelity design problem is a physical problem for which at least two mathematical or computational models exist: $f(\mathbf{x})$ with $\mathbf{c}(\mathbf{x})$ and $\hat{f}(\hat{\mathbf{x}})$ with $\hat{\mathbf{c}}(\hat{\mathbf{x}})$. The parameterization of a model is the set of design variables \mathbf{x} or $\hat{\mathbf{x}}$ used as inputs to the model. A variable-parameterization problem is a variable-fidelity problem in which each of the models has a different parameterization, meaning that for the same physical design, $\mathbf{x} \neq \hat{\mathbf{x}}$. A mapping is a method for linking the design variables in a variable-parameterization problem. Given a set of design variables in one parameterization, it provides a set of design variables in another parameterization. The dimension of a model is the number of design variables. A variable-dimensional problem is a variable-parameterization problem in which each of the models has a different dimension: that is, where $n \neq \hat{n}$.

This paper first presents the TRMM framework, including the SQP-like constrained optimization method. It then outlines design variable mapping and specifically introduces corrected space mapping. It then presents the results of two example problems: a

wing planform design and the design of a batlike flapping wing. Finally, it draws some conclusions.

Trust-Region Model Management

Surrogates can be incorporated into optimization by using a formal model-management strategy. One such strategy is a TRMM framework [26]. TRMM imposes limits on the amount of optimization performed using the low-fidelity model, based on a quantitative assessment of that model's predictive capability. TRMM developed from the classical trust-region optimization method based on quadratic Taylor series models [27].

TRMM methods are provably convergent to an optimum of the high-fidelity model, as long as the two models satisfy a number of conditions, including that the low-fidelity model is corrected to be at least first-order consistent with the high-fidelity model. The complete list of conditions and a proof are available in [16]. The general approach in TRMM is to solve a sequence of optimization subproblems using only the low-fidelity model, with an additional constraint that requires the solution of the subproblem to lie within a specified trust region. The radius of the trust region is adaptively managed on each subproblem iteration using a merit function to quantitatively assess the predictive capability of the low-fidelity model.

The SQP-like method is modified from [10]. It is similar to sequential quadratic programming (SQP) in that on each subproblem it minimizes a surrogate of the Lagrangian subject to linear approximations to the high-fidelity constraints.

The Lagrangian is defined as

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda^T \mathbf{c}(\mathbf{x}) \quad (2)$$

where λ is the vector of Lagrange multipliers.

The SQP-like TRMM algorithm is as follows:

1) Choose an initial point \mathbf{x}^0 and an initial trust-region radius $\Delta^0 > 0$. Choose an initial approximation λ^0 to the Lagrange multipliers. Set $k = 0$. Choose constants $\eta > 0$, $0 < c_1 < 1$, $c_2 > 1$, $0 < r_1 < r_2 < 1$, and $\Delta^* \geq \Delta^0$.

2) Create a surrogate \tilde{L} for the Lagrangian L . The surrogate must be at least first-order consistent with the Lagrangian at the center of the trust region: that is,

$$\tilde{L}^k(\mathbf{x}^k, \lambda^k) = L^k(\mathbf{x}^k, \lambda^k) \quad (3)$$

$$\nabla_{\mathbf{x}} \tilde{L}^k(\mathbf{x}^k, \lambda^k) = \nabla_{\mathbf{x}} L^k(\mathbf{x}^k, \lambda^k) \quad (4)$$

In this work, the surrogate for the Lagrangian is created by using separate surrogates for the objective and each constraint: that is,

$$\tilde{L}(\mathbf{x}, \lambda) = \tilde{f}(\mathbf{x}) + \lambda^T \tilde{\mathbf{c}}(\mathbf{x}) \quad (5)$$

The surrogates \tilde{f} for f and \tilde{c}_j for each c_j are created using mapping and correction on the low-fidelity model \hat{f} and \hat{c}_j . Mapping and correction are described in the next section.

3) Solve the k th trust-region subproblem

$$\begin{aligned} & \min_{\mathbf{s}} \tilde{L}^k(\mathbf{x}^k + \mathbf{s}, \lambda^k) \\ & \text{subject to } \mathbf{c}^k(\mathbf{x}^k + \mathbf{s}) + \nabla_{\mathbf{x}} \mathbf{c}^k(\mathbf{x}^k + \mathbf{s})^T (\mathbf{x} - \mathbf{x}^k) \leq 0 \| \mathbf{s} \|_2 \leq \Delta^k \end{aligned} \quad (6)$$

and set the trial step \mathbf{s}^k to the minimizing step. The method for solving the problem must result in a step satisfying the fraction of Cauchy decrease condition [16] on the subproblem.

4) Compute $f(\mathbf{x}^k + \mathbf{s}^k)$ and $c(\mathbf{x}^k + \mathbf{s}^k)$. The acceptance criteria uses dominance, a concept borrowed from multiobjective optimization [28]. A point \mathbf{x}^1 dominates a point \mathbf{x}^2 if both of the following conditions are satisfied:

$$f(\mathbf{x}^1) \leq f(\mathbf{x}^2), \quad \| \mathbf{c}^+(\mathbf{x}^1) \|_2 \leq \| \mathbf{c}^+(\mathbf{x}^2) \|_2 \quad (7)$$

where $\mathbf{c}^+(\mathbf{x})$ is a vector with elements defined by

$$c_i^+(\mathbf{x}) = \max(0, c_i(\mathbf{x})) \quad (8)$$

A filter is a set of points, none of which dominate any other. In a filter method, the initial filter is empty. The trial point $\mathbf{x}^k + \mathbf{s}^k$ is accepted and added to the filter if it is not dominated by any point in the filter. If any element of the filter dominates the trial point, the trial point is rejected and not added to the filter. Significant detail on filter methods is available in chapter 5 of [16]. If the trial step is accepted, $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{s}^k$. If the trial step is rejected, $\mathbf{x}^{k+1} = \mathbf{x}^k$.

5) Define the trust-region ratio

$$\rho^k = \frac{L(\mathbf{x}^k, \lambda^k) - L(\mathbf{x}^k + \mathbf{s}^k, \lambda^k)}{\bar{L}(\mathbf{x}^k, \lambda^k) - \bar{L}^k(\mathbf{x}^k + \mathbf{s}^k, \lambda^k)} \quad (9)$$

Set

$$\Delta^{k+1} = \begin{cases} c_1 \|\mathbf{s}^k\| & \text{if } \rho^k < r_1, \\ \min(c_2 \Delta^k, \Delta^*) & \text{if } \rho^k > r_2, \\ \|\mathbf{s}^k\| & \text{otherwise} \end{cases} \quad (10)$$

If both the numerator and the denominator in Eq. (9) are zero or very small, and the step is accepted using the filter rules, the trust-region size is increased. If only the denominator is very small, the trust-region size is decreased.

6) Calculate new values for the Lagrange multipliers. The Lagrange multipliers are updated by solving the nonnegative least-squares constraint problem:

$$\min_{\lambda} \|\nabla_{\mathbf{x}} f(\mathbf{x}_0^k) + \sum_{i \in S} \lambda_i \nabla_{\mathbf{x}} c_i(\mathbf{x}_0^k)\|_2^2 \quad \text{subject to } \lambda \geq 0 \quad (11)$$

where S is the set of active constraints, using the nonnegative least-squares algorithm in Sec. 23.3 of [29]. The solution to this problem is λ^{k+1} . Increment k by 1 and go to step 2.

Mapping

SBO methods have until now been applicable only to models in which both the high-fidelity model $f(\mathbf{x})$ [$\mathbf{c}(\mathbf{x})$] and the low-fidelity model $\hat{f}(\hat{\mathbf{x}})$ [$\hat{\mathbf{c}}(\hat{\mathbf{x}})$] are defined over the same space $\mathbf{x} = \hat{\mathbf{x}}$. To use a low-fidelity model with a different number of design variables from the high-fidelity function to be optimized, it is necessary to find a relationship between the two sets of design vectors: that is, $\hat{\mathbf{x}} = P(\mathbf{x})$. Then $\hat{f}(P(\mathbf{x}))$ is corrected to a surrogate for $f(\mathbf{x})$, and $\hat{\mathbf{c}}(P(\mathbf{x}))$ is corrected to a surrogate for $\mathbf{c}(\mathbf{x})$. The optimization algorithm then calculates trial steps in the high-fidelity space. Another option is to calculate steps in the low-fidelity space and to correct $\hat{f}(\hat{\mathbf{x}})$ to a surrogate for $f(Q(\hat{\mathbf{x}}))$ and $\hat{\mathbf{c}}(\hat{\mathbf{x}})$ to a surrogate for $\mathbf{c}(Q(\hat{\mathbf{x}}))$. The latter option requires constraints on the Jacobian of the mapping to ensure that the projection of the gradient is finite for a finite gradient [30] and will not be addressed here.

In some cases, this design space mapping can be obvious and problem-specific. For instance, if the high- and low-fidelity models are the same set of physical equations but on a fine grid and a coarse grid and the design vectors in each case are geometric parameters defined on those grids, the low-fidelity design vector can be a subset of the high-fidelity design vector or the high-fidelity design vector can be an interpolation of the low-fidelity design vector. However, in other problems, there is no obvious mathematical relationship between the design vectors. In this case, an empirical mapping is needed. One example of such a problem is the flapping-flight problem described in this paper. Another is the multifidelity supersonic business jet problem used by Choi et al. [31]. Because then-existing SBO methods cannot be applied to problems in which the low- and high-fidelity models use different design variables, Choi et al. used the two models sequentially, optimizing first using the low-fidelity model, with kriging corrections applied, and using the result of that optimization as a starting point for optimization using the high-fidelity model. This also required an additional step of

manually mapping the low-fidelity optimum to the high-fidelity space to provide a starting point for high-fidelity optimization.

Space Mapping

Space mapping, first introduced by Bandler et al. [32], links the high- and low-fidelity models through their input parameters. The goal of space mapping is to vary the input parameters to the low-fidelity model to match the output of the high-fidelity model. In microwave circuit design, for which space mapping was first developed, it is often appropriate to make corrections to the input of a model, rather than to its output.

The first space-mapping-based optimization algorithm used a linear mapping between the high- and low-fidelity design spaces. It used a least-squares solution of the linear equations resulting from associating corresponding data points in the two spaces. Space-mapping optimization consists of optimizing in the low-fidelity space and inverting the mapping to find a trial point in the high-fidelity space. New data points near the trial point are then used to construct the mapping for the next iteration. This process is repeated until no further progress is made. Although this method can result in substantial improvement (as demonstrated by several design problems, most in circuit design [33], but some in other disciplines [34]), it is not provably convergent to even a local minimum of the high-fidelity space. In fact, although improvement in the high-fidelity model is often possible when the low-fidelity model is similar to the high-fidelity model, it is not guaranteed.

Space mapping was further improved with the introduction of aggressive space mapping [35]. Aggressive space mapping descends more quickly toward the optimum than space mapping, but requires the assumptions that the mapping between the spaces is bijective and that it is always possible to find a set of low-fidelity design vectors that, when fed into the low-fidelity model, provide an output almost identical to the high-fidelity model evaluated at any given high-fidelity design vector. It also requires that the design variables are the same dimension in both spaces. Because the method does not ensure first-order accuracy, the proofs of convergence of trust-region methods do not extend to those methods using space mapping. However, Madsen and Søndergaard [36] developed a provably convergent algorithm by using a hybrid method in which the surrogate is a convex combination of the space-mapped low-fidelity function and a Taylor series approximation to the high-fidelity function.

The space-mapping examples available in the literature consider only the case in which the design vectors have the same length. Therefore, this work expands it to include the variable-parameterization case, including when the design vectors are not the same length.

In space mapping, a particular form is assumed for the relationship P between the high- and low-fidelity design vectors. This form is described by some set of space-mapping parameters, contained here in a vector \mathbf{p} , that are found by solving an optimization problem:

$$\mathbf{p} \in \arg \min_{\mathbf{p}} \sum_{i=1}^q (\|\beta(\mathbf{x}^i) - \hat{\beta}(P(\mathbf{x}^i, \mathbf{p}))\|_2) \quad (12)$$

This optimization problem seeks to minimize the difference between some high-fidelity function $\beta(\mathbf{x})$ and the corresponding low-fidelity function $\hat{\beta}(\hat{\mathbf{x}}) = \hat{\beta}(P(\mathbf{x}, \mathbf{p}))$ over a set of q sample points \mathbf{x}^i , where \mathbf{x}^i denotes the i th sample point. Both the choice of sample points and the particular form of the mapping P are left to the implementation.

Corrections

As mentioned in the preceding SQP-like algorithm, provable convergence of the TRMM requires at least first-order consistency between the high-fidelity model and the surrogate model. This can be accomplished using corrections. Corrections can be additive or multiplicative; this work uses additive corrections. Although only first-order corrections are required, quasi-second-order corrections have been shown to accelerate convergence of a TRMM [18] and are therefore used in this work.

For some low-fidelity function $\hat{\beta}(\hat{\mathbf{x}})$, the corresponding high-fidelity function $\beta(\mathbf{x})$, and a mapping $\hat{\mathbf{x}} = P(\mathbf{x})$, the k th additive-corrected surrogate is defined as

$$\tilde{\beta}^k(\mathbf{x}) = \hat{\beta}(P(\mathbf{x})) + A^k(\mathbf{x}) \quad (13)$$

To obtain quasi-second-order consistency between $\tilde{\beta}^k(\mathbf{x}^k)$ and $\beta(\mathbf{x}^k)$, we define the correction function $A^k(\mathbf{x})$ using a quadratic Taylor series expansion of the difference $\mathcal{A}(\mathbf{x})$ between the two functions β and $\hat{\beta}$ about the point \mathbf{x}^k :

$$A^k = \mathcal{A}(\mathbf{x}^k) + \nabla_{\mathbf{x}} \mathcal{A}(\mathbf{x}^k)^T (\mathbf{x} - \mathbf{x}^k) + \frac{1}{2} (\mathbf{x} - \mathbf{x}^k)^T \nabla_{\mathbf{x}}^2 \mathcal{A}(\mathbf{x}^k) (\mathbf{x} - \mathbf{x}^k) \quad (14)$$

The elements in this expansion are calculated using

$$\mathcal{A}(\mathbf{x}^k) = \beta(\mathbf{x}^k) - \hat{\beta}(P(\mathbf{x}^k)) \quad (15)$$

$$\frac{\partial \mathcal{A}(\mathbf{x}^k)}{\partial x_p} = \frac{\partial \beta}{\partial x_p}(\mathbf{x}^k) - \sum_{j=1}^{\hat{n}} \frac{\partial \hat{\beta}}{\partial \hat{x}_j}(P(\mathbf{x}^k)) \frac{\partial \hat{x}_j}{\partial x_p}, \quad p = 1, \dots, n \quad (16)$$

$$\begin{aligned} \frac{\partial^2 \mathcal{A}(\mathbf{x}^k)}{\partial x_p \partial x_q} &= H_{pq}^k - \sum_{j=1}^{\hat{n}} \left(\frac{\partial \hat{\beta}}{\partial \hat{x}_j}(P(\mathbf{x}^k)) \frac{\partial^2 \hat{x}_j}{\partial x_p \partial x_q} \right. \\ &\quad \left. + \sum_{\ell=1}^{\hat{n}} \hat{H}_{j\ell}^k \frac{\partial \hat{x}_j}{\partial x_p} \frac{\partial \hat{x}_\ell}{\partial x_q} \right), \quad p = 1, \dots, n, \quad q = 1, \dots, n \end{aligned} \quad (17)$$

where x_p denotes the p th element of the vector \mathbf{x} , \mathbf{H}^k is the Broyden–Fletcher–Goldfarb–Shanno (BFGS) approximation to the Hessian matrix of the high-fidelity function β at \mathbf{x}^k , $\hat{\mathbf{H}}^k$ is the BFGS approximation to the Hessian matrix of the low-fidelity function $\hat{\beta}$ at $P(\mathbf{x}^k)$, and H_{pq}^k denotes the pq th element of the matrix \mathbf{H}^k .

For each subproblem k , Eq. (15) computes the difference between the value of the high-fidelity function and the low-fidelity function at the center of the trust region. Using the chain rule, Eq. (16) computes the difference between the gradient of the high-fidelity function and the gradient of the low-fidelity function at the center of the trust region, in which the gradients are computed with respect to the high-fidelity design vector \mathbf{x} . The second term in Eq. (16) therefore requires the Jacobian of the mapping, $\partial \hat{x}_j / \partial x_p$. Similarly, Eq. (17) computes the difference between the BFGS approximation of the Hessian matrices of the high-fidelity and low-fidelity functions at the center of the trust region. Once again, derivatives are required with respect to \mathbf{x} and are computed using the chain rule.

Corrected Space Mapping

Because space mapping does not provide provable convergence within a TRMM framework, but any surrogate that is first-order accurate does, one approach is to correct the space-mapping framework to at least first order. This can be done with the corrections described previously. However, if the input parameters are first selected to match the output function at some number of control points and a correction is subsequently applied, it is likely that the correction will unnecessarily distort the match performed in the space-mapping step. This can be resolved by performing the space mapping and correction steps simultaneously, which is achieved by embedding the correction within the space mapping.

This concept is illustrated in Fig. 1, which shows the available data points (\circ) and the center of the trust region (\times). The dotted curve is a cubic function found with a least-squares fit to the available data. It provides no consistency at the trust-region center. The dashed curve shows the result of adding a linear additive correction to that fit to enforce first-order accuracy at the center of the trust region. The local correction distorts the global data fitting. The solid curve is also a

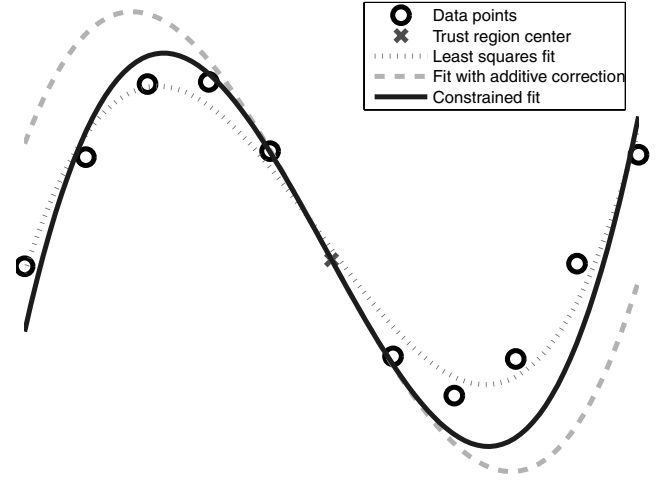


Fig. 1 Demonstration of simultaneous vs sequential data fitting and enforcement of first-order accuracy.

cubic function, generated by first enforcing first-order accuracy at the center and then performing a least-squares fit with the remaining degrees of freedom. This last curve is more globally accurate than the sequential fitting and correction steps.

Using this concept, corrected space mapping performs the space mapping and correction steps simultaneously. That is, it incorporates a correction, and with the remaining degrees of freedom it performs the best match possible to the control points by varying the input mapping.

The corrected space-mapping optimization problem is

$$\mathbf{p}^k \in \arg \min_{\mathbf{p}} \sum_{i=1}^q \|\beta(\mathbf{x}^i) - \tilde{\beta}^k(P(\mathbf{x}^i, \mathbf{p}))\|_2 \quad (18)$$

Equation (18) is the same as Eq. (12) with $\hat{\beta}$, the uncorrected low-fidelity function, replaced by $\tilde{\beta}^k$, the corrected surrogate to the high-fidelity function on the k th subproblem. The optimization problem (18) seeks to minimize the difference between the high-fidelity and surrogate objective functions over a set of k sample points \mathbf{x}^i , where \mathbf{x}^i denotes the i th sample (or control) point. Both the choice of sample points and the particular form of the mapping P are left to the implementation. The correction, because it depends on the Jacobian and Hessian matrices of the mapping, must be updated for each new value of \mathbf{p} .

In the implementation employed in this work, the sample points used in Eq. (12) are the previous q accepted steps in the TRMM algorithm ($\mathbf{x}^{k-q+1}, \dots, \mathbf{x}^k$) at which high-fidelity function values are already available. A linear relationship is chosen for the mapping P :

$$\hat{\mathbf{x}} = P(\mathbf{x}) = \mathbf{M}\mathbf{x} + \mathbf{b} \quad (19)$$

where \mathbf{M} is a matrix with $\hat{n} \times n$ elements, and \mathbf{b} is a vector of length \hat{n} for a total of $\hat{n} \times (n + 1)$ space-mapping parameters. It should be noted that other forms of the mapping could also be used. The space-mapping parameters must be determined at each iteration of the TRMM method by solving the optimization problem (18). This additional optimization problem in $\hat{n} \times (n + 1)$ dimensional space adds computational cost that increases with the number of design variables. However, in many applications, such as computational fluid dynamic problems, this additional algorithm overhead is significantly less than the cost of a function evaluation. Thus, the algorithm provides net computational savings. This is illustrated in the example problems.

Example Problems

This paper presents two constrained example problems: a wing planform design problem and the design of a batlike flapping-wing vehicle. Previous work has addressed unconstrained problems,

including two variations of the Rosenbrock problem and an airfoil design problem [37], and has compared a number of constrained trust-region optimization strategies on the Barnes problem and the wing design problem presented subsequently [38].

Wing Design Problem

The first example design problem is the constrained planform design of a wing. The wingspan is constant at 10 m and the angle of attack is set to a constant value of 0.5 deg. The quarter-chord is unswept. The objective function is the coefficient of induced drag, and the lift of the wing is constrained from below. This problem is linked to a more complex aircraft design problem in which the lift must be equal to the weight and the drag of the aircraft is minimized. The high-fidelity design variables specify the chord at each of 10 evenly distributed points along the wing. The optimization problem is given by

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) &= C_{Di}(\mathbf{x}) \\ \text{subject to } c(\mathbf{x}) &= (0.2 - A(\mathbf{x})C_L(\mathbf{x})) \leq 0 \\ 0.01 \leq x_j &\leq 10, \quad j = 1, \dots, 10 \end{aligned} \quad (20)$$

where C_{Di} is the coefficient of induced drag, C_L is the coefficient of lift, A is the wing area, \mathbf{x} is a vector containing the chord design variables, and x_j is the chord at the j th spanwise station.

The high-fidelity code uses a vortex-lattice method using a zero-thickness, constant-collocation, doublet-lattice model [39]. By imposing Neumann boundary conditions and a zero-spanwise-vorticity trailing-edge Kutta condition, the wing and wake surface doublet strengths can be uniquely determined. The discretization is performed using quadrilateral panels with uniform chordwise spacing and cosine spanwise refinement. A standard Trefftz-plane analysis [39] is used to compute both the lift and induced drag. A single analysis of the high-fidelity code takes approximately 90 s on a 2.4 GHz Intel Pentium 4 desktop workstation.

The low-fidelity code uses lifting-line theory. A standard lifting-line method following Ashley and Landahl [40] has been implemented to compute both the lift and the induced drag. Because it assumes that the wing-bound vorticity can be approximated by a single spanwise line vortex, the lifting-line method is less accurate than the high-fidelity vortex-lattice method. The low-fidelity design variables are also chords, but in this case, at only three points that are again equally spaced from the root to the tip. This problem is therefore an example of variable parameterization: the low-fidelity design space is of lower dimension than the high-fidelity design space. A single analysis of the low-fidelity code takes approximately 30 ms on a 2.4 GHz desktop workstation.

For both the high-fidelity and low-fidelity models, gradients were calculated using centered finite differences. The evaluations required for these finite differences are included in the count of the number of function calls. The benchmark method is SQP. The multifidelity method is the SQP-like trust-region method in conjunction with corrected space mapping. The eight previous iterates were used as the control points. Before the accumulation of eight iterates, all previous iterates were used as the control points. The linear space mapping of Eq. (19) was used. The initial design was a rectangular wing and was feasible with the lift constraint inactive.

Given infinite degrees of freedom, the planform that yields minimum induced drag is elliptic. The optimization problem is expected to find a distribution that most closely approximates an elliptical planform. Figure 2 shows the initial and final planforms. Figure 3 shows the objective function value and constraint violation of each method versus the number of high-fidelity function calls. Both the benchmark method and the multifidelity method converged to the same near-elliptic design. The high-fidelity SQP method took 1344 high-fidelity function calls, including those required to calculate gradients, to achieve the optimum design, with an objective within 10^{-5} of the best design found, with a constraint violation less than 10^{-6} . The multifidelity method found the optimum, using the same criteria, in 319 high-fidelity function calls. As the cost of the

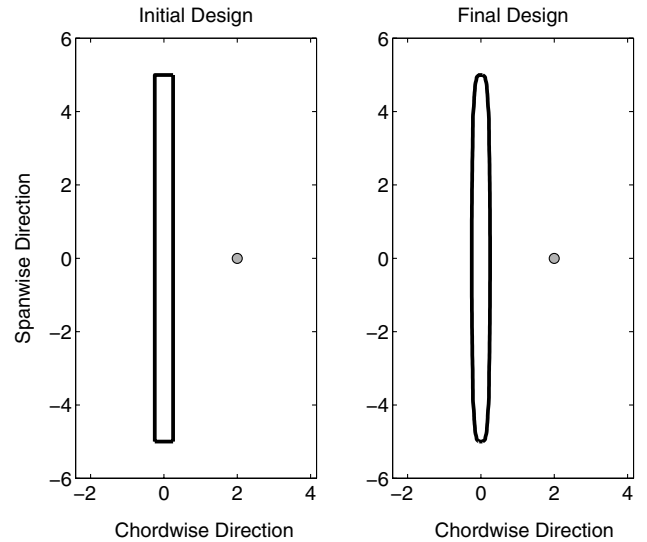


Fig. 2 Contour plot of kriging-generated response surface of trapped region size as a function of elliptical cutout geometry.

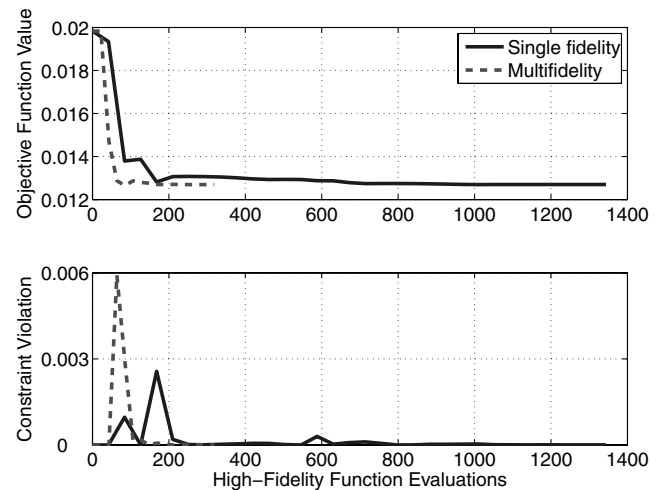


Fig. 3 Objective function value and constraint violation of each of a single-fidelity SQP method and an SQP-like multifidelity method using corrected space mapping.

low-fidelity model was negligible, the total computational time was reduced from approximately 34 h to approximately 8 h, a savings of 76%.

Flapping-Flight Problem

The final constrained design problem aims to explore biologically inspired flapping flight. Heaving wings and airfoils are commonly used in nature as a form of force production and have gained the attention of many researchers, both to further understand biological flight and to develop bioinspired aircraft [41–44].

Bats have been studied in a wind tunnel to understand the mechanics of bat flight [45]. Figure 4 shows a bat outfitted with markers to track bat motion in flight in a wind tunnel. Researchers postulate that bats' physiological structure and flight motion are optimized for minimum power input at a specified forward velocity. To test that hypothesis, a model problem was formulated to investigate the minimization of power to generate thrust, using leading-edge compliance. The goal of this problem is to examine passive strategies for minimizing the heaving motion power input for a given thrust output. Bat flight is modeled using a heaving airfoil with a passive load-alleviation leading-edge torsional spring. Although birds and bats in nature use more complex forms of both active and passive control, this optimization problem is a first step in

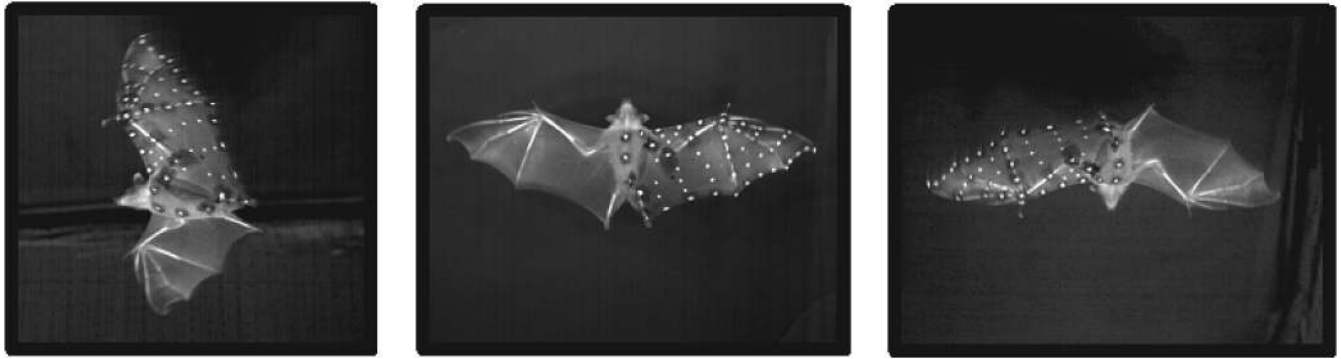


Fig. 4 Marker locations on a *Cynopterus brachyotis*, or short-nosed fruit bat, outfitted for testing in the Harvard Concord Field Station Wind Tunnel (figure courtesy of Kenneth Breuer and Sharon Swartz, Brown University).

the determination of the optimal passive strategy. A similar problem of minimizing the power requirements for a swimming robot using passive springs has been investigated using analytic approaches [46].

Problem Description

The high-fidelity model of the design problem has eight design parameters. The first is a spring constant. This represents the structural compliance of the bat wing, modeled as a single leading-edge torsional spring providing passive load alleviation. The second design parameter is the flapping frequency. The next three are amplitudes and lags of each of three harmonics of flapping motion. Table 1 shows the eight design variables, their units, their lower and upper bounds, their start values, and their final values in each of the single-fidelity and multifidelity runs. The optimization problem is

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) = P_{\text{in}}(\mathbf{x}) \quad \text{subject to } c(\mathbf{x}) = (0.2 - C_T(\mathbf{x})) \leq 0 \\ LB \leq x_j \leq UB, \quad j = 1, \dots, 8 \end{aligned} \quad (21)$$

where the input power P_{in} , the thrust coefficient C_T , and the elements of the design parameter vector

$$\mathbf{x} = (K, \omega, A_0, A_1, A_2, \phi_0, \phi_1, \phi_2)^T$$

are as described in Table 1. The design variables were scaled to improve the numerical conditioning of the problem.

High-Fidelity Model

The high-fidelity solver is a two-dimensional, unsteady, linear strength, source-doublet formulation [39,47]. An advantage of using panel-method approximations in an unsteady setting is that it requires neither remeshing nor moving-body formulations (such as arbitrary Lagrange–Euler formulations of the Navier–Stokes equations). The unsteady forces and moments were computed by integrating the airfoil surface pressure, computed using the unsteady form of the Bernoulli equation [39]. To correct the results for viscous

effects, a simple quasi-steady drag-polar approximation was used. Additionally, a simple stall penalty scheme consisting of a quartic drag penalty on airfoil incidences over a specified value was also incorporated to ensure that the angle of incidence of the airfoil remained in the nonseparated regime. Although the viscous model is not as rigorous as one that depends on the unsteady motion of the geometry (as would be the case for an integral-boundary-layer method [48–50]), the incorporation of a simple viscous correction yields more realistic computations than an inviscid formulation. The passive structural load alleviation and airfoil rotation were accomplished by modeling the airfoil as a mass and a leading-edge torsional spring. The following moment-balance equation was enforced strongly for each time step at the leading edge of the airfoil:

$$I\ddot{\theta} + K\theta + mx_{\text{cg}}\ddot{h} - M_{\text{aero}} = 0 \quad (22)$$

where I is the moment of inertia about the leading edge of the airfoil, θ is the angle of the wing, K is the spring constant of the torsional spring at the leading edge, m is the mass of the wing in the wake direction, x_{cg} is the x position of the center of gravity of the wing in the wake direction, h is the vertical position of the wing (and thus \ddot{h} is its vertical acceleration), and M_{aero} is the moment due to aerodynamic forces.

A low-Reynolds-number HT-13 airfoil was used as the input geometry. The vertical heaving motion was described by a third-order series of harmonic functions as follows:

$$Z(t) = \sum_{m=0}^2 A_m \cos(2\pi(m+1)\omega t + \phi_m) \quad (23)$$

where $Z(t)$ is the z position, defined along a vertical axis, of the airfoil in time; t is time; and A_i , ϕ_i ($i = 0, \dots, 2$), and ω are as defined in Table 1. The horizontal velocity is constant at

$$U(t) = U_{\infty} = 5 \text{ m/s} \quad (24)$$

The high-fidelity code requires approximately 85 s on a 2.4 GHz desktop workstation for a single evaluation.

Table 1 Design variables and results for the flapping-wing problem

Symbol	Name	Unit	Lower bound	Upper bound	Initial	Final, single fidelity	Final, multifidelity
<i>Design variables</i>							
K	Spring constant	(N · m)/rad	0.0001	0.25	0.005	0.0136	0.00949
ω	Flapping frequency	rad/s	10	120	20	11.4643	13.41
A_0	Amplitude of harmonic 0	m	0	0.3	0.2	0.3	0.3
A_1	Amplitude of harmonic 1	m	0	0.15	0	7.93×10^{-5}	2.20×10^{-3}
A_2	Amplitude of harmonic 2	m	0	0.15	0	4.82×10^{-3}	0.0145
ϕ_0	Lag of harmonic 0	rad	$-\pi$	π	0	-3.058	-2.148
ϕ_1	Lag of harmonic 1	rad	$-\pi$	π	0	0	-0.208
ϕ_2	Lag of harmonic 2	rad	$-\pi$	π	0	-1.943	0.0626
<i>Constraint</i>							
C_T	Coefficient of thrust	—	0.2	—	0.1076	0.2	0.2
<i>Objective function</i>							
P_{in}	Power input	W	—	—	0.5461	0.8966	0.9049

Low-Fidelity Model

The low-fidelity model has three design variables: a spring constant, a flapping frequency, and a flapping amplitude. Only one flapping harmonic is included. The low-fidelity analysis uses a simplified representation of the wing as a point airfoil. The calculations are quasi-steady. That is, the airfoil changes in position and angle of attack over time, but the aerodynamic equations are steady. The airfoil is assumed to be massless. The moment balance around the leading edge is therefore

$$K\theta + M_{c/4}(\alpha) - \frac{c}{4}L_{\text{aero}}(\alpha) = 0 \quad (25)$$

where $M_{c/4}$ is the aerodynamic moment about the quarter-chord, c is the chord, and L_{aero} is the aerodynamic lift. Using a thin-airfoil-theory approximation, the lift is

$$L_{\text{aero}} = 2\pi\alpha \quad (26)$$

and the moment of the symmetric airfoil about the quarter-chord is

$$M_{c/4}(\alpha) = 0 \quad (27)$$

The angle of attack α can be approximated as the difference between the relative airflow direction (which can be determined from the forward and heaving velocities) and the angular deflection θ of the airfoil. As a result, the spring deflection in Eq. (25) is the only remaining unknown and is easily determined if the terms in the equation are approximated using harmonic representations. Once the spring deformations are known, the time-varying lift and thrust forces are computed using the magnitude of the force prediction and the angle of attack of the airfoil. In addition to the ideal aerodynamic forces, a simple quasi-steady drag-polar viscous correction was implemented postsolution. This was added to the model to ensure that adversely high angles of attack or velocities did not provide unrealistically high thrust values. Figure 5 shows the flapping motion of the wing and the resultant forces. The low-fidelity code requires approximately 20 ms on a 2.4 GHz desktop workstation to evaluate a single design.

Results

Two methods were run: a single-fidelity SQP method as a benchmark and the SQP-like multifidelity method with corrected space mapping. As in the previous problem, the gradients of the objectives and of the constraints were computed using centered finite differences. The function evaluations required for these finite difference calculations are included in the total function count. The methods converged to two different local minima. The Karush–Kuhn–Tucker condition was satisfied at both minima. This result highlights the point that the multifidelity method is guaranteed to converge to an optimum of the high-fidelity problem, but not necessarily the global optimum or the same (possibly local) optimum found by the single-fidelity method. The presence of multiple local optima was verified by starting the single-fidelity method at the point found by the multifidelity method. For both methods, the amplitude of the first harmonic (that is, the flapping motion at the base frequency) was set to its upper bound. No other design parameters were set to their bounds.

The flapping frequencies were similar: 11.46 rad/s for the single-fidelity method and 13.41 rad/s for the multifidelity method. These correspond to Strouhal numbers of 0.218 and 0.256, respectively. These are both within the 0.2 to 0.4 range found in birds, bats, and fish in nature [41,51]. The multifidelity method found a design with an input power coefficient of 0.9049, and the single-fidelity method found a design with an input power coefficient of 0.8966. The difference between the two values is within 1% and below the predicted accuracy of the high-fidelity analysis.

The objective function and constraint violations are shown in Fig. 6, with the number of high-fidelity function calls on the x axis. Using the criterion that the constraint violation is less than 10^{-6} and the objective function is within 10^{-6} of its ultimate converged value, the single-fidelity method required 3125 high-fidelity function calls

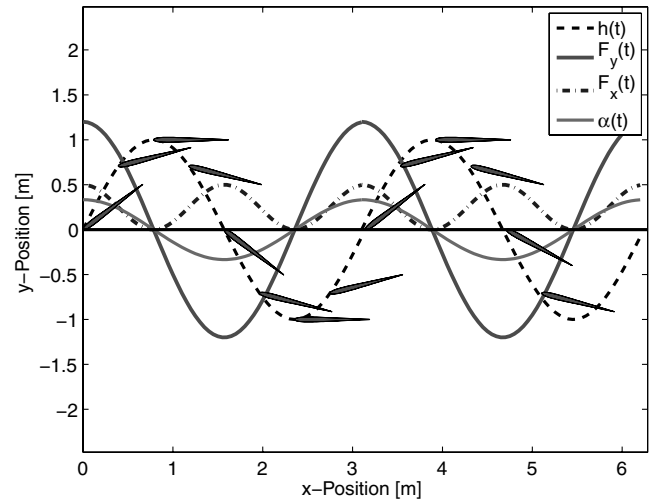


Fig. 5 Results of the low-fidelity analysis of a flapping wing showing the airfoil position, angle of attack, and resultant forces in the x and y directions.

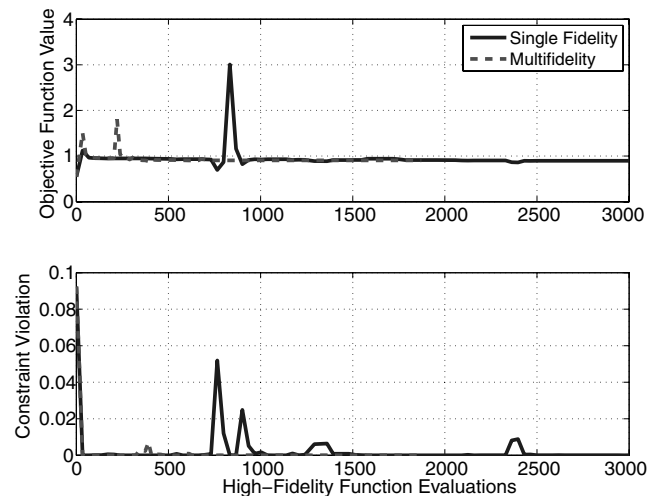


Fig. 6 Objective function value and constraint violation for the flapping-wing problem with the single-fidelity method and with the SQP-like multifidelity method with corrected space mapping.

and the multifidelity method required 1617. Because they did not converge to the same minimum, it is difficult to make a direct comparison between the numbers. The multifidelity method found a solution with a slightly higher objective function value with 48.2% fewer high-fidelity function evaluations. The multifidelity method required 77 h to run and the single-fidelity method required 140 h. This is a savings of approximately 45% in time. The difference in ratios between the function evaluation metric and the time metric are explained by the low-fidelity function calls used by the multifidelity method and the additional multifidelity algorithm overhead.

Conclusions

Reducing the computational cost of optimization for expensive simulation-based design problems is an important research challenge in engineering optimization. This paper aimed to extend SBO methods to variable-parameterization multifidelity problems: that is, problems for which multiple models exist and use different sets of design variables. It presented a new mapping method, corrected space mapping, and applied it using an SQP-like TRMM. The method is applicable to optimization problems with computationally expensive high-fidelity simulations and cheaper low-fidelity simulations, including those with large numbers of design variables.

Its utility is highest when the computational cost of the high-fidelity model is much higher than that of the low-fidelity model.

On a constrained wing design problem, the method achieved 76% savings in high-fidelity function evaluations, reducing the time required for optimization from 34 to 8 h. On a bat-flight design problem, it achieved approximately 45% time savings, although it converged to a different local minimum than did the benchmark. This mapping method shows promise for a wide range of problems with variable-parameterization hierarchical models.

Acknowledgments

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed-Martin Company, for the U.S. Department of Energy under contract DE-AC04-94AL85000. The authors acknowledge the financial support of the Computational Engineering Program of the Singapore and Massachusetts Institute of Technology alliance and the Computer Science Research Institute at Sandia National Laboratories. David Willis developed the aerodynamic codes used for both example problems.

References

- [1] Venter, G., Haftka, R., and Starnes, J. H., "Construction of Response Surface Approximations for Design Optimization," *AIAA Journal*, Vol. 36, No. 12, Dec. 1998, pp. 2242–2249.
doi:10.2514/2.333
- [2] Myers, R., and Montgomery, D., *Response Surface Methodology*, 2nd ed., Wiley, New York, 2002.
- [3] Sacks, J., Welch, W., Mitchell, T., and Wynn, H., "Design and Analysis of Computer Experiments," *Statistical Science*, Vol. 4, No. 4, 1989, pp. 409–436.
doi:10.1214/ss/1177012413
- [4] Powell, M., "Radial Basis Functions for Multivariable Interpolation," *Algorithms for Approximation*, edited by J. Mason, and M. Cox, Oxford Univ. Press, New York, 1987, pp. 141–167.
- [5] Mullur, A., and Messac, A., "Extended Radial Basis Functions: More Flexible and Effective Metamodeling," *AIAA Journal*, Vol. 43, No. 6, 2005, pp. 1306–1315.
doi:10.2514/1.11292
- [6] Mullur, A., and Messac, A., "Metamodeling Using Extended Radial Basis Functions: A Comparative Approach," *Engineering with Computers*, Vol. 21, No. 3, May 2006, pp. 203–217.
doi:10.1007/s00366-005-0005-7
- [7] Sirovich, L., "Turbulence and the Dynamics of Coherent Structures Part 1: Coherent Structures," *Quarterly of Applied Mathematics*, Vol. 45, No. 3, Oct. 1987, pp. 561–571.
- [8] Banerjee, J., "Exact Modal Analysis of an Idealised Whole Aircraft Using Symbolic Computation," *The Aeronautical Journal*, Vol. 104, No. 1035, May 2000, pp. 247–255.
- [9] Forrester, A. I., Bressloff, N. W., and Keane, A. J., "Optimization Using Surrogate Models and Partially Converged Computational Fluid Dynamics Simulations," *Proceedings of the Royal Society of London, Series A: Mathematical and Physical Sciences*, Vol. 462, Mar. 2000, pp. 2177–2204.
doi:10.1098/rspa.2006.1679
- [10] Alexandrov, N., Lewis, R., Gumbert, C., Green, L., and Newman, P., "Optimization with Variable-Fidelity Models Applied to Wing Design," 38th Aerospace Sciences Meeting and Exhibit, Reno, NV, AIAA Paper 2000-0841, Jan. 2000.
- [11] Alexandrov, N. M., Nielsen, E., Lewis, R., and Anderson, W., "First-Order Model Management with Variable-Fidelity Physics Applied to Multi-Element Airfoil Optimization," 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Long Beach, CA, AIAA Paper 2000-4886, Sept. 2000.
- [12] Jones, D., "A Taxonomy of Global Optimization Methods Based on Response Surfaces," *Journal of Global Optimization*, Vol. 21, No. 4, 2001, pp. 345–383.
doi:10.1023/A:1012771025575
- [13] Queipo, N. V., Haftka, R. T., Shyy, W., Goel, T., Vaidyanathan, R., and Tucker, P. K., "Surrogate-Based Analysis and Optimization," *Progress in Aerospace Sciences*, Vol. 41, No. 1, Jan. 2005, pp. 1–28.
doi:10.1016/j.paerosci.2005.02.001
- [14] Powell, M. J. D., "A New Algorithm for Unconstrained Optimization," *Nonlinear Programming*, edited by J. Rosen, O. Mangasarian, and K. Ritter, Academic Press, New York, 1970, pp. 31–65.
- [15] Lewis, R. M., "A Trust Region Framework for Managing Approximation Models in Engineering Optimization," 6th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Design, Bellevue, WA, AIAA Paper 96-4101, 1996.
- [16] Conn, A. R., Gould, N. I., and Toint, P. L., *Trust-Region Methods, MPS/SIAM Series on Optimization*, Society for Interactive and Applied Mathematics, Philadelphia, 2000.
- [17] Fletcher, R., Leyffer, S., and Toint, P. L., "On the Global Convergence of an SLP-Filter Algorithm," Dept. of Mathematics, Facultés Univ. Notre Dame de la Paix, TR 98/13, Namur, Belgium, 1998.
- [18] Eldred, M., Giunta, S., and Collis, S., "Second-Order Corrections for Surrogate-Based Optimization with Model Hierarchies," 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Albany, NY, AIAA Paper 2004-4457, 2004.
- [19] Yuan, Y., "A Review of Trust Region Algorithms for Optimization," Inst. of Computational Mathematics, TR 99-038, Beijing, 1999.
- [20] Booker, A., Dennis, J., Frank, P., Serafini, D., Torczon, V., and Trosset, M., "A Rigorous Framework for Optimization of Expensive Functions by Surrogates," *Structural and Multidisciplinary Optimization*, Vol. 17, No. 1, No. 1, 1999, pp. 1–13.
doi:10.1007/BF01197708
- [21] Audet, C., Dennis, J., Moore, D. W., Booker, A., and Frank, P. D., "A Surrogate-Model-Based Method for Constrained Optimization," 8th AIAA/USAF/NASA/ASSMO Symposium on Multidisciplinary Analysis and Optimization, Long Beach, CA, AIAA Paper 2000-4891, Sept. 2000.
- [22] Audet, C., and Dennis, J., "A Pattern Search Filter Method for Nonlinear Programming Without Derivatives," *SIAM Journal on Optimization*, Vol. 14, No. 4, 2004, pp. 980–1010.
doi:10.1137/S105262340138983X
- [23] Rodriguez, J., Renaud, J., and Watson, L., "Trust Region Augmented Lagrangian Methods for Sequential Response Surface Approximation and Optimization," *Journal of Mechanical Design*, Vol. 120, No. 1, Mar. 1998, pp. 58–66.
doi:10.1115/1.2826677
- [24] El-Alem, M., "A Global Convergence Theory for Dennis, El-Alem, and Maciel's Class of Trust-Region Algorithms for Constrained Optimization Without Assuming Regularity," *SIAM Journal on Optimization*, Vol. 9, No. 4, 1999, pp. 965–990.
doi:10.1137/S1052623497331762
- [25] Sadjadi, S. J., and Ponnambalam, K., "Advances in Trust Region Algorithms for Constrained Optimization," *Applied Numerical Mathematics*, Vol. 29, No. 3, 1999, pp. 423–443.
doi:10.1016/S0168-9274(98)00105-6
- [26] Alexandrov, N., Dennis, J., Lewis, R., and Torczon, V., "A Trust-Region Framework for Managing the Use of Approximation Models in Optimization," *Structural and Multidisciplinary Optimization*, Vol. 15, No. 1, Feb. 1998, pp. 16–23.
doi:10.1007/BF01197433
- [27] Dennis, J. E., "A Brief Introduction to Quasi-Newton Methods," *Numerical Analysis*, Vol. 22, American Mathematical Society, Providence, RI, 1978, pp. 19–52.
- [28] Srinivas, N., and Deb, K., "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms," *Evolutionary Computation*, Vol. 2, No. 3, 1994, pp. 221–248.
doi:10.1162/evco.1994.2.3.221
- [29] Lawson, C., and Hanson, R., *Solving Least-Squares Problems*, Prentice-Hall, New York, 1974, p. 161.
- [30] Robinson, T., "Surrogate-Based Optimization using Multifidelity Models with Variable Parameterization," Ph.D. thesis, Massachusetts Institute of Technology, May 2007.
- [31] Choi, S., Alonso, J., Kim, S., and Kroo, I., "Two-Level Multi-Fidelity Design Optimization Studies for Supersonic Jets," 43rd AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, AIAA Paper 2005-531, Jan. 2005.
- [32] Bandler, J., Biernacki, R., Chen, S., Grobelny, P., and Hemmers, R., "Space Mapping Technique for Electromagnetic Optimization," *IEEE Transactions on Microwave Theory and Techniques*, Vol. 42, No. 12, Dec. 1994, pp. 2536–2544.
doi:10.1109/22.339794
- [33] Bakr, M. H., Bandler, J. W., Georgieva, N., and Madsen, K., "A Hybrid Aggressive Space Mapping Algorithm for EM Optimization," *IEEE MTT-S International Microwave Symposium Digest*, Vol. 1, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, 1999, pp. 265–268.
- [34] Bandler, J., Biernacki, R., and Chen, S., "Fully Automated Space Mapping Optimization of 3D Structures," *Proceedings of the IEEE MTT-S International Microwave Symposium*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, June 1996, pp. 753–756.
- [35] Bandler, J., Biernacki, R., Chen, S., Hemmers, R., and Madsen, K.,

- “Electromagnetic Optimization Exploiting Aggressive Space Mapping,” *IEEE Transactions on Microwave Theory and Techniques*, Vol. 43, No. 12, Dec. 1995, pp. 2874–2882.
doi:10.1109/22.475649
- [36] Madsen, K., and Søndergaard, J., “Convergence of Hybrid Space Mapping Algorithms,” *Optimization and Engineering*, Vol. 5, No. 2, June 2004, pp. 145–156.
doi:10.1023/B:OPTE.0000033372.34626.49
- [37] Robinson, T., Eldred, M., Willcox, K., and Haimes, R., “Strategies for Multifidelity Optimization with Variable Dimensional Hierarchical Models,” 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Newport, RI, AIAA Paper 2006-1819, May 2006.
- [38] Robinson, T., Willcox, K., Eldred, M., and Haimes, R., “Multifidelity Optimization for Variable-Complexity Design,” 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Portsmouth, VA, AIAA Paper 2006-7114, Sept. 2006.
- [39] Katz, J., and Plotkin, A., *Low-Speed Aerodynamics*, 2nd ed., Cambridge Univ. Press, Cambridge, England, U.K., 2001.
- [40] Ashley, H., and Landahl, M., *Aerodynamics of Wings and Bodies*, Dover, New York, 1985.
- [41] Triantafyllou, M., Triantafyllou, G., and Gopalkrishnan, R., “Wake mechanics for thrust generation in oscillating foils,” *Physics of Fluids A*, Vol. 3, No. 12, 1991, pp. 2835–2837.
doi:10.1063/1.858173
- [42] Mueller, T. (ed.), *Fixed and Flapping Wing Aerodynamics for Micro Air Vehicles*, Progress in Aeronautics and Astronautics, AIAA, Reston, VA, 2001.
- [43] Wang, Z., “Vortex Shedding and Frequency Selection in Flapping Flight,” *Journal of Fluid Mechanics*, Vol. 410, 2000, pp. 323–341.
doi:10.1017/S0022112099008071
- [44] Hall, K., Pigott, S., and Hall, S., “Power Requirements for Large-Amplitude Flapping Flight,” *Journal of Aircraft*, Vol. 35, No. 3, 1998, pp. 352–361.
doi:10.2514/2.2324
- [45] Tian, X., Iriarte-Diaz, J., Middleton, K., Galvao, R., Israeli, E., Roemer, A., Sullivan, A., Song, A., Swartz, S., and Breuer, K., “Direct Measurements of the Kinematics and Dynamics of Bat Flight,” *Bioinspiration and Biomimetics*, Vol. 1, No. 4, Dec. 2006, pp. S10–S18.
doi:10.1088/1748-3182/1/4/S02
- [46] Harper, K., Berkemeier, M., and Grace, S., “Decreasing the Energy Costs of Swimming Robots Through Passive Elastic Elements,” *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, 1997, pp. 1839–1844.
- [47] Morino, L., and Kuo, C., “Subsonic Potential Aerodynamics for Complex Configurations: A General Theory,” *AIAA Journal*, Vol. 12, No. 2, 1974, pp. 191–197.
doi:10.2514/3.49191
- [48] Drela, M., and Giles, M., “Viscous-Inviscid Analysis of Transonic and Low Reynolds Number Airfoils,” *AIAA Journal*, Vol. 25, No. 10, Oct. 1987, pp. 1347–1355.
doi:10.2514/3.9789
- [49] Drela, M., “XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils,” *Low Reynolds Number Aerodynamics*, edited by T. Mueller, Springer-Verlag, New York, June 1989.
- [50] Eppler, R., and Somers, D., “Low Speed Airfoil Design and Analysis, Advanced Technology Airfoil Research: Volume 1,” NASA CP 2045, 1979.
- [51] Taylor, G., Nudda, R., and Thomas, A., “Flying and Swimming Animals Cruise at a Strouhal Number Tuned for High Power Efficiency,” *Nature*, Vol. 425, Oct. 2003, pp. 707–711.
doi:10.1038/nature02000

A. Messac
Associate Editor