AIAA JOURNAL
Vol. 56, No. 8, August 2018

Bayesian-Enhanced Low-Fidelity Correction Approach to Multifidelity Aerospace Design

C. Corey Fischer* and Ramana V. Grandhi†
Wright State University, Dayton, Ohio 45435
and
Philip S. Beran‡
U.S. Air Force Research Laboratory, Wright–Patterson Air Force Base, Ohio 45433

DOI: 10.2514/1.J056529

This work discusses a novel approach, Bayesian-inspired multifidelity optimization, for solving complex optimization requiring computationally expensive simulations. Bayesian-inspired multifidelity optimization aims to minimize computational cost in achieving accurate high-fidelity results through the use of low-fidelity (computationally cheaper) models in combination with a surrogate correction model. A novel Bayesian hybrid bridge function was developed to serve as the low-fidelity correction technique. This Bayesian hybrid bridge function is a Bayesian weighted average of two standard bridge functions, additive and multiplicative. The correction technique is implemented in parallel with a modified trust region model management optimization scheme. It is shown that optimization on the corrected low-fidelity model converges to the same local optimum as optimization on the high-fidelity model in fewer high-fidelity model. This technique is developed to reach that optimum in fewer high-fidelity function evaluations than traditional optimization, ultimately reducing computational cost. This work also extends the low-fidelity correction optimization beyond the traditional bifidelity optimization to that of optimization with multiple-fidelity objective and constraint functions. The proposed solution technique allows for use of commercial optimizers and is demonstrated on three separate problems. The first demonstration is on a one-dimensional analytical test case in which the Bayesian correction is shown. Then, the ability to handle multiple fidelities in both the objective and constraint functions is presented and discussed using a two-dimensional analytical test case. An airfoil shape optimization problem is then used to illustrate the effectiveness of this approach in an engineering design scenario in which the various fidelities arise as a difference in computational physics. It is shown via these demonstrations that implementation of this Bayesian low-fidelity correction optimization approach results in convergence to a high-fidelity optimum at a reduced computational cost in comparison to traditional optimization techniques.

Nomenclature

a, b, c, r, s, t	=	generic, nondimensional coefficients for Branin and/or Hicks–Henne function
C	=	generic, nondimensional coefficient
f	=	generic function
g	=	generic constraint function
LB, UB	=	optimization lower and upper bounds
n	=	generic number of design points
r	=	trust region model management nondimensional decision bounds
S	=	generic size
s	=	trust region model management nondimensional scaling value
w	=	nondimensional weighting coefficient
x	=	generic design variable
$\boldsymbol{x}$	=	generic design point (array of all design variables)
y	=	nondimensional airfoil outer mold line
β, γ	=	additive or multiplicative correction
Δ	=	generic maximum step size
ϵ	=	generic convergence criteria
ρ	=	trust region model management nondimensional accuracy criteria
σ	=	generic standard deviation
ϕ	=	corrected low-fidelity function
ψ	=	likelihood function

Subscripts

Add, Mult	=	additive or multiplicative term
bot, top	=	bottom or top airfoil surface
$c, *$	=	center or optimal point
Drag, Lift	=	drag or lift coefficient
H, L	=	high- or low-fidelity function
i, j	=	ith or jth design variable
max, min	=	maximum or minimum value
MLE	=	maximum likelihood estimator
TR	=	value specific to trust region
WR	=	value specific to weight region

Superscripts

k	=	kth iteration value
0	=	initial value
$\hat{}$	=	estimated value
$\sim$	=	surrogate function

Presented as Paper 2017-0133 at the 58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Grapevine, TX, 9–13 January 2017; received 14 July 2017; revision received 8 January 2018; accepted for publication 27 March 2018; published online 25 June 2018.

*Ph.D. Candidate, Department of Mechanical and Materials Engineering; fischer.36@wright.edu. Student Member AIAA.
†Distinguished Research Professor, Department of Mechanical and Materials Engineering. Fellow AIAA.
‡Principal Research Aerospace Engineer, Air Vehicles Directorate. Associate Fellow AIAA.

I. Introduction

A. Motivation

NUMERICAL optimization methodologies are growing in popularity for use in aircraft design as physical interactions are understood and system requirements increase in complexity and demand. These methodologies offer the potential of increased system

performance through enabling numerous design options to be explored systematically. While optimization methods have traditionally been predominant in the latter stages of a design process (preliminary and detailed), there is a growing interest and need for the use of higher-fidelity physics in the earlier stages of design (conceptual) [1]. However, simulation models based on these higher-fidelity physics tend to have higher computational cost in comparison to their lower-fidelity counterparts. Therefore, optimization, which typically requires a large number of model evaluations, can be prohibitively expensive given the higher computational cost of these physics-based models. The traditional method for reducing the computational cost inherently present in optimization is to employ surrogate-based optimization (SBO) techniques. This paper presents an alternative/modified technique that leverages high-fidelity response data to correct low-fidelity models for use in an SBO environment.

The aforementioned need for high-fidelity physics to be brought into the design cycle at an earlier stage of the design process is evident in the design of next-generation military aircraft, both manned and unmanned. These aircraft demand increased capabilities in speed, range, survivability, mission versatility, and reliability [2]. To satisfy these demands, one must achieve synergy between aircraft constituent subsystems including, among others, propulsion, structures, flight controls, and materials. This necessary synergy, and resulting maximum platform performance, is only attainable through the use of a truly integrated design process. Such a process, along with the desire to identify and exploit beneficial coupling within the physics of the design domain, inherently requires leveraging higher-fidelity computational simulations among various disciplines early in the design process. This stands contrary to conventional conceptual design practices that use handbooks, spreadsheets, and legacy information. However, it is currently unclear as to when it is appropriate and/or necessary to bring in higher-fidelity simulation models, or even experimental data, that will provide the best benefit to the design process.

The conceptual and preliminary design stages of aircraft design have traditionally been two separate, time-intensive design phases. However, the concept of dialable/multifidelity design is one in which the gap between these two phases is bridged by introducing physics into the design process at an earlier stage than traditionally employed. This introduction of physics at an earlier stage eliminates the need for separate conceptual and preliminary design phases and consequently reduces the total design time.

The concept of multifidelity design does, however, pose certain obstacles such as determining how and when to dial or switch between different fidelity models. This paper explores the ability of applying an adjustment factor to the response of a low-fidelity model so as to predict the true system response taken to be the response obtained from a high-fidelity simulation throughout an optimization routine. A surrogate model is constructed for the purpose of determining an adjustment factor given any design point (thus a function of design variables) using information of previous high- and low-fidelity simulations from previous optimization iterations. In doing so, sensitivity information of the high-fidelity simulation model can be estimated through a combination of sensitivity information for the adjustment factor surrogate model and low-fidelity models. It is shown that optimization on the high-fidelity as well as adjusted low-fidelity models converges to the same local optimum, whereas optimization on an adjusted low-fidelity model does so in an order of magnitude fewer high-fidelity function evaluations.

In this research, an adaptation to traditional trust region model management schemes has been developed and employed within an optimization routine designed to facilitate the decision-making process of determining when to use higher-fidelity response information. This is implemented in parallel with surrogate modeling techniques, such as Kriging, for the purpose of constructing a model of adjustment factors. This surrogate adjustment factor model is then used to correct the low-fidelity model.

B. Multifidelity Optimization

Numerous heuristic techniques have been used to optimize a high-fidelity function using lower-fidelity information. We consider heuristic multifidelity optimization (MFO) approaches to be approaches that generally converge in practice to an optimum of the high-fidelity function, but in which there is no formal mathematical proof or guarantee. These methods vary from problem-specific necessities to rigorous methods that compute the probability of finding an improved high-fidelity function value. Examples of problem-specific multifidelity approaches include adding global response surface corrections to low-fidelity models [3,4]; using the low-fidelity function gradient as the optimization direction, but performing the line search with the high-fidelity function values [5]; creating a response surface using both high- and low-fidelity analysis results [6]; and running higher-fidelity models when two or more lower-fidelity models disagree [7]. In contrast, we consider a nonheuristic method to be one in which, given a set of requirements for the initial design(s) and behavior of the high- and low-fidelity functions, there is a mathematical guarantee that with enough time the multifidelity method will find a high-fidelity optimum [8]. Nonheuristic multifidelity methods may converge more slowly than single-fidelity methods, but nonetheless they are guaranteed to work eventually. Our discussion of MFO methods focuses on both heuristic and nonheuristic methods that are broadly applicable and likely to find a high-fidelity optimum for general problems.

In MFO, global and local approaches define two sides of a coin on which research focuses. Global methods search the entire feasible domain for the best design, whereas local methods attempt to find the nearest design that has better performance than all other designs in that neighborhood. Some approaches combine an augmented Lagrangian [9], exact penalty method, constraint filtering, or barrier with a pattern-search or a simplex method. Other methods use linear interpolation of both the objective function and the constraint [10]. Global methods have the benefit that they typically do not require estimates of the high-fidelity functions' gradients. This is important because frequently a high-fidelity functions' gradient is unavailable and cannot be estimated accurately. However, given the extreme advancements in sensitivity calculation approaches, this paper makes the assumption that gradient information is not beyond the scope of attainment [11]. A detriment to using global optimization methods is that they typically require considerably more high-fidelity evaluations than local methods. So, there is clearly a need for both types. This paper focuses on local methods that may use global approach ideology and methodology such as enhanced surrogate modeling techniques.

C. Open Issues in Multifidelity Optimization

This paper only considers local optimization and does not attempt to find the globally optimal design due to the challenges addressed. For local multifidelity optimization, four major challenge themes have emerged: 1) how to rigorously and effectively combine multiple low-fidelity models to best predict the high-fidelity function behavior, 2) knowing when is it appropriate to use theh high-fidelity model and when is it safe for lower-fidelity models to be used to drive the design process, 3) how to find a high-fidelity optimum in the presence of potentially computationally expensive function evaluations, and 4) how best to use gradient information such that surrogate models use as much information as is known about the high-fidelity function. These challenges are also significant in multidisciplinary system design, as they may emerge within the design of each subsystem and at the integrated-system level.

D. Multifidelity Basics

This paper outlines a multifidelity optimization process, which enables the use of low-fidelity models to alleviate the computational cost of high-fidelity models while maintaining a desired high-fidelity level of accuracy through a novel Bayesian-influenced low-fidelity correction depicted in Fig. 1. This multifidelity optimization process entails the implementation of a trust region model management (TRMM) methodology alongside a Bayesian hybrid bridge function (BHBF), which requires the use of a surrogate modeling technique. Because of spherical step size constraints imposed by the TRMM methodology, this paper adopts the use of a gradient-enhanced

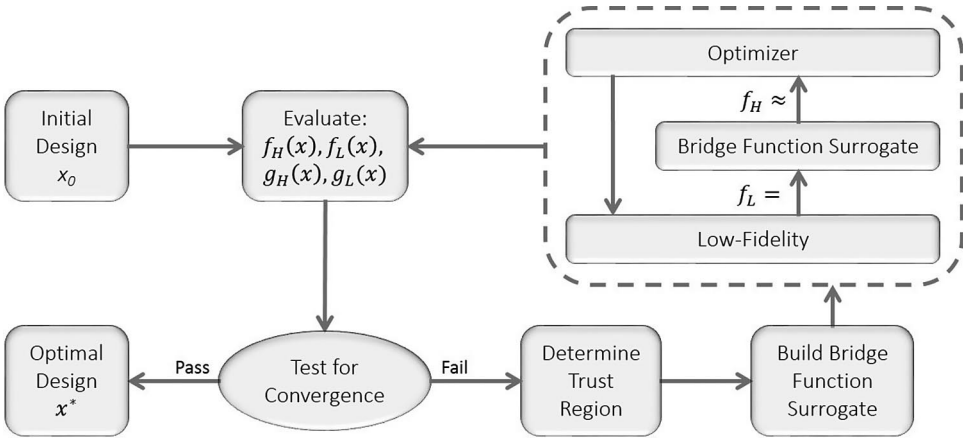


Fig. 1 Multifidelity optimization flow chart.

Kriging (GEK) surrogate modeling technique. The TRMM scheme is used to define a trust region, over which surrogates of both high- and low-fidelity models are constructed. These surrogates are constructed using all design points located within the trust region bounds. Note that high- and low-fidelity simulation responses are available at all design points from previous optimization iterations (subproblem optimization on corrected low fidelity defining a single iteration). Next, the separate GEK models are used in the calculation of additive and multiplicative adjustment factors. These models are then used within the BHBF methodology to correct the low-fidelity model. Finally, gradient-based optimization is performed on the corrected low-fidelity model. The process described is outlined by Fig. 1 and posed in Algorithm 1. The individual components are described in further detail throughout this paper. Note that in the event that multiple fidelities are available for both the objective and constraint (s) a special consideration must be imposed. This issue is addressed in Sec. III.

II. Trust Region Model Management

SBO methods can be implemented in a multitude of ways, one of the most common being a TRMM framework. This framework consists of an iterative process on which a surrogate of the desired model is optimized during each iteration, rather than the full model. The concept behind the model management scheme is the adaptive nature in which the size of the trust region is adjusted based upon the accuracy of the surrogate model. This work modifies traditional approaches to fit the needs of a self-contained multifidelity optimization methodology.

Given a scenario in SBO for which the surrogate has been accurate at predicting the model response over previous iterations, the trust region grows. However, conversely, the trust region shrinks if the surrogate is inaccurate. The TRMM framework can be applied to both constrained and unconstrained optimization problems [5,12].

To solve the nonlinear programming problem, the TRMM method solves a sequence of trust region optimization subproblems. Note that this approach assumes consistent design variables between the high-fidelity model and the surrogate. The surrogate is permitted to change from iteration to iteration, taking on the form of

$$\begin{aligned} &\text{minimize:} \quad \tilde{f}^k(\boldsymbol{x}) \\ &\text{subject to:} \quad \|\boldsymbol{x} - \boldsymbol{x}_c^k\| \le \Delta^k \end{aligned} \tag{1}$$

for the kth subproblem.

In Eq. (1), $\tilde{f}^k$ denotes the kth surrogate model, $\boldsymbol{x}_c^k$ is the solution to the previous subproblem optimization (and center point for the trust region of the current subproblem), Δ^k is the size of the trust region for subproblem k, and the initial trust region size Δ^0 is user selected. The solution to the kth subproblem is denoted by $\boldsymbol{x}_*^k$. After each of the k iterations in the TRMM framework, the predicted step is validated by computing the trust region ratio ρ^k using [13]

$$\rho^k = \frac{f(\boldsymbol{x}_c^k) - f(\boldsymbol{x}^k)}{f(\boldsymbol{x}_c^k) - \tilde{f}(\boldsymbol{x}_*^k)} \tag{2}$$

This is the ratio of the actual improvement of the objective or constraint function to the improvement predicted by optimization on the surrogate model, thus a measure of the accuracy of the surrogate. This ratio measures the performance of the surrogate model by finding new iterates that improve the objective while satisfying constraints via ensuring model accuracy. Traditionally, this value has been used to determine step size acceptance as well as the size of the next trust region using the logic of

Algorithm 1 Multifidelity optimization

Require: $w^0 = (w^0_{\text{Add}}, w^0_{\text{Mult}})$, $\text{TRS}^0 = (1/4)(\boldsymbol{x}_{\max} - \boldsymbol{x}_{\min})/2$, $s = (s_1, s_2, s_3)$, $r = (r_1, r_2, r_3, r_4)$

1: Define initial point $\boldsymbol{x}^0$ and global bounds $\boldsymbol{x}_{\max}$ and $\boldsymbol{x}_{\min}$.
2: Evaluate low- and high-fidelity models at $\boldsymbol{x}^0$ (function and sensitivity values).
3: Perform TRMM.
4: Build low-fidelity $[\tilde{f}_L(\boldsymbol{x})]$ and high-fidelity $[\tilde{f}_H(\boldsymbol{x})]$ surrogate models on TR data.
5: Define additive $[\tilde{\beta}(\boldsymbol{x}) = \tilde{f}_H(\boldsymbol{x}) - \tilde{f}_L(\boldsymbol{x})]$ and multiplicative $(\tilde{\gamma}(\boldsymbol{x}) = \tilde{f}_H(\boldsymbol{x})/\tilde{f}_L(\boldsymbol{x}))$ adjustment factor models using high- and low-fidelity surrogates.
6: Define additive $[\tilde{f}_{H,\beta}(\boldsymbol{x}) = \tilde{\beta}(\boldsymbol{x}) + f_L(\boldsymbol{x})]$ and multiplicative $[\tilde{f}_{H,\gamma}(\boldsymbol{x}) = \tilde{\gamma}(\boldsymbol{x}) f_L(\boldsymbol{x})]$ correctedlow-fidelity models.
7: Calculate hybrid Bayesian weighting coefficients $[w^k = (w_{\text{Add}}, w_{\text{Mult}})]$ using weight region data.
8: Define corrected low-fidelity model $\tilde{f}_H = w_{\text{Add}}\tilde{f}_{H,\beta}(\boldsymbol{x}) + w_{\text{Mult}}\tilde{f}_{H,\gamma}(\boldsymbol{x})$.
9: Perform subproblem optimization (obtain optimal point $\boldsymbol{x}^* = \boldsymbol{x}^k$).
10: Evaluate low- and high-fidelity models at $\boldsymbol{x}^k$ (function and sensitivity values).
11: Calculate stopping criteria $\epsilon = (f_H(\boldsymbol{x}^{k-1}) - f_H(\boldsymbol{x}^k))/f_H(\boldsymbol{x}^{k-})$.
if $\epsilon < \epsilon^*$ **then** stop
else return to step 3 and proceed

$$\begin{cases} \rho^k \le 0 & \text{reject step size and shrink trust region size by } 1/2 \\ 0 < \rho^k \le r_1 & \text{accept step size and shrink trust region size by } 1/2 \\ r_1 < \rho^k < r_2 & \text{accept step size and maintain trust region size} \\ r_2 \le \rho^k & \text{accept step size and increase trust region size by 2} \end{cases} \tag{3}$$

Therefore, this framework calls the high-fidelity function once per iteration to check the accuracy of the surrogate and performs the actual optimization on the surrogate model [9].

This paper adapts the previously mentioned derivation in three distinct aspects. First, as has been discovered, the trust region model management methodology was sensitive to the user-defined accuracy bounds as well as the scaling factor imposed on the trust region given a certain accuracy level. This sensitivity has proven to be problem dependent, thus making it difficult to define a good standard that was pervasive to a large range of design problems. As was the case, this paper focused on addressing these issues as described by

$$\begin{cases} \rho^k \le r_1 & \text{scale trust region size by } s_1 \\ r_1 < \rho^k \le r_2 & \text{scale trust region size by } s_2 \\ r_2 < \rho^k < r_3 & \text{scale trust region size by} s_3 \\ r_3 \le \rho^k & \text{scale trust region size by } s_4 \end{cases} \tag{4}$$

and

$$\begin{cases} \rho^k \le r_1 & \text{accept step size} \\ \rho^k > r_1 & \text{reject step size} \end{cases} \tag{5}$$

Second, the traditional TRMM framework has been altered by removing the constraint of step size in Eq. (1) and replacing this constraint by adaptive lower and upper bounds on the trust region imposed as side bounds in the optimization. Finally, this method was further adapted by permitting the user to select the rate at which the trust region either shrank or grows so as to allow for a more robust approach as defined in Eq. (4). The new decision rules are defined in

$$\text{TRS} = \left| \frac{\text{UB}^{k-1} - \text{LB}^{k-1}}{2} \right| \tag{6}$$

and

$$\begin{cases} \rho^k \le r_1 & \text{LB}^k = \max(\boldsymbol{x}_{\min}, \boldsymbol{x}_c^{k-1} - s_1 S_{\text{TR}}), \text{UB}^k = \max(\boldsymbol{x}_{\min}, \boldsymbol{x}_c^{k-1} + s_1 S_{\text{TR}}) \\ r_1 < \rho^k \le r_2 & \text{LB}^k = \max(\boldsymbol{x}_{\min}, \boldsymbol{x}_*^{k-1} k - s_2 S_{\text{TR}}), \text{UB}^k = \max(\boldsymbol{x}_{\min}, \boldsymbol{x}_*^{k-1} + s_2 S_{\text{TR}}) \\ r_2 < \rho^k < r_3 & \text{LB}^k = \max(\boldsymbol{x}_{\min}, \boldsymbol{x}_*^{k-1} - s_3 S_{\text{TR}}), \text{UB}^k = \max(\boldsymbol{x}_{\min}, \boldsymbol{x}_*^{k-1} + s_3 S_{\text{TR}}) \\ r_3 \le \rho^k & \text{LB}^k = \max(\boldsymbol{x}_{\min}, \boldsymbol{x}_*^{k-1} - s_4 S_{\text{TR}}), \text{UB}^k = \max(\boldsymbol{x}_{\min}, \boldsymbol{x}_*^{k-1} + s_4 S_{\text{TR}}) \end{cases} \tag{7}$$

In these equations, note that $0 < s_1 < s_2 < 1 \le s_3 < s_4$, where it has been found that it is best for $s_1 s_4 \ne 1$ from previous work by Fischer and Grandhi [14]. Also note that $0 \le r_1 < r_2 < r_3 < 1$, where it has been found that $0 \le r_1 \le 10^{-4}$, $10^{-3} \le r_3 \le 0.25$, and $0.75 \le r_3 < 1$. These previous works by Fischer and Grandhi as well as Eldred and Dunlavy [4] are experienced based and formulated off of literature recommendations.

Note that the lower and upper bounds of the side constraints imposed on the design variables $\boldsymbol{x}$ are never permitted to extend beyond the initial bounds of the high-fidelity optimization problem ($\boldsymbol{x}_{\min}$ and $\boldsymbol{x}_{\max}$). This formulation still accounts for the scenario in which the subproblem optimization step is rejected ($\rho^k \le r_1$) and thus shrinks the trust region size but centers this trust region about the same center as the previous iteration. This trust region model management process described previously has been summarized in Algorithm 2. Note that in this algorithm S_{TR} denotes the trust region size; WRS denotes the weight region size for use in the Bayesian posterior updating technique addressed later in this paper; and LB_{TR}, UB_{TR}, LB_{WR}, and UB_{WR} represent the trust region lower and upper bounds as well as weight region lower and upper bounds, respectively.

Algorithm 2 TRMM

Require: $f_H(\boldsymbol{x}_c^k), f_H(\boldsymbol{x}_*^k), \tilde{f}_H(\boldsymbol{x}_*^k), s = (s_1, s_2, s_3), r = (r_1, r_2, r_3, r_4)\, S_{\text{TR}}^{k-1}$, WRS

1: Calculate accuracy criterion
$\rho^k = (f_H(\boldsymbol{x}_c^k) - f_H(\boldsymbol{x}_*^k))/(f_H(\boldsymbol{x}_c^k) - \tilde{f}_H(\boldsymbol{x}_*^k))$.
2: Determine acceptance.
if $\rho^k \le r_1$ **then** reject step $\rightarrow \boldsymbol{x}_c^k = \boldsymbol{x}_c^{k-1}$
else accept step $\rightarrow \boldsymbol{x}_c^k = \boldsymbol{x}_*^{k-1}$
3: Determine trust region scale value.
if $\rho^k \le r_1$
$s^k = s_1$
else if $r_1 < \rho^k \le r_2$
$s^k = s_2$
else if $r_2 < \rho^k \le r_3$
$s^k = s_3$
else
$s^k = s_4$
4: Update trust region size $S_{\text{TR}}^k = s^k S_{\text{TR}}^{k-1}$.
5: Define trust region and weight region bounds:
$(\text{LB}_{\text{TR}}^k = \boldsymbol{x}_c^k - S_{\text{TR}}^k, \text{UB}_{\text{TR}}^k = \boldsymbol{x}_c^k + S_{\text{TR}}^k)$
$(\text{LB}_{\text{WR}}^k = \boldsymbol{x}_c^k - \text{WRS} * S_{\text{TR}}^k, \text{UB}_{\text{WR}}^k = \boldsymbol{x}_c^k + \text{WRS} * S_{\text{TR}}^k)$

III. Multifidelity Objective and Constraint(s)

As mentioned previously, it is very likely that an engineering design problem will exhibit multifidelity objective and constraint definitions. This scenario poses somewhat of an obstacle in defining correction models as well as handling said models in the TRMM framework. This problem has only been addressed in the sense of converting the constrained optimization problem to an unconstrained problem through the use of a Lagrange multiplier technique common within optimization [9]. The pitfall to this approach is that we no longer look at the objective and constraint(s) separately. This poses a problem when trying to develop two or more BHBFs, one for the objective and one for each of the constraints, and handling them in the TRMM framework. This also prevents the use of commercial nonlinear optimization packages such as those available in MATLAB®, Python, and Design Optimization Toolbox. Therefore, this paper aims to address these issues. First, as outlined in Algorithm 3, and shown in Fig. 2, a multifidelity design technique has been developed to handle a scenario in which multiple fidelities exist for the objective and at least one of the constraints (or a minimum of two constraints). This scenario is handled by constructing separate correction models for the objective (if multiple fidelity objective) and each constraint for which multiple fidelities are defined. Next, these models are handled by the TRMM methodology in which an accuracy criterion is calculated for the corrected low-fidelity objective and the corrected low-fidelity constraint(s) using Eq. (2). Upon calculating an accuracy criterion for the objective and the constraint(s), the lowest (least accurate) accuracy criterion is passed

Algorithm 3 Multifidelity optimization

Require: $w^0 = (w^0_{\text{Add}}, w^0_{\text{Mult}})$, $S^0_{\text{TR}} = (1/4)(\boldsymbol{x}_{\max} - \boldsymbol{x}_{\min})/2$, $s = (s_1, s_2, s_3)$, $r = (r_1, r_2, r_3, r_4)$

1: Define initial point $\boldsymbol{x}^0$ and global bounds $\boldsymbol{x}_{\max}$ and $\boldsymbol{x}_{\min}$.
2: Evaluate low- and high-fidelity models at $\boldsymbol{x}^0$ (function and sensitivity values) for objective and constraint(s).
3: Perform TRMM for objective and constraint(s).
4: Least accurate correction model [objective and constraint(s)] dictate TRMM acceptance and scaling (used for accuracy criterion in TRMM).
5: Build low-fidelity $[\tilde{f}_L(\boldsymbol{x})]$ and high-fidelity $[\tilde{f}_H(\boldsymbol{x})]$ surrogate models on TR data for objective and constraint(s) separately.
6: Define additive $[\tilde{\beta}(\boldsymbol{x}) = \tilde{f}_H(\boldsymbol{x}) - \tilde{f}_L(\boldsymbol{x})]$ and multiplicative $(\tilde{\gamma}(\boldsymbol{x}) = \tilde{f}_H(\boldsymbol{x})/\tilde{f}_L(\boldsymbol{x}))$ adjustment factor models using high- and low-fidelity surrogates for objective and constraint(s) separately.
7: Define additive $[\tilde{f}_{H,\beta}(\boldsymbol{x}) = \tilde{\beta}(\boldsymbol{x}) + f_L(\boldsymbol{x})]$ and multiplicative $[\tilde{f}_{H,\gamma}(\boldsymbol{x}) = \tilde{\gamma}(\boldsymbol{x}) f_L(\boldsymbol{x})]$ corrected low-fidelity models for objective and constraint(s) separately.
8: Calculate hybrid Bayesian weighting coefficients $[w^k = (w_{\text{Add}}, w_{\text{Mult}})]$ using weight region data for objective and constraint (s) separately.
9: Define corrected low-fidelity model $\tilde{f}_H = w_{\text{Add}}\tilde{f}_{H,\beta}(\boldsymbol{x}) + w_{\text{Mult}}\tilde{f}_{H,\gamma}(\boldsymbol{x})$ for objective and constraint(s) separately.
10: Perform subproblem optimization (obtain optimal point $\boldsymbol{x}^* = \boldsymbol{x}^k$).
11: Evaluate low- and high-fidelity models at $\boldsymbol{x}^k$ (function and sensitivity values).
12: Calculate stopping criteria $\epsilon = (f_H(\boldsymbol{x}^{k-1}) - f_H(\boldsymbol{x}^k))/f_H(\boldsymbol{x}^{k-})$.
if $\epsilon < \epsilon^*$ **then** stop
else return to step 3 and proceed

to Eqs. (6) and (7). This ensures that the trust region movement and resizing are based upon the least accurate model, thus ensuring the minimal desired level of accuracy is always satisfied. Note that the Lagrange multiplier approach would not work in this scenario because an extremely accurate model could outweigh an inaccurate model. This would lead to an instance in which a correction model is assumed to be accurate, thus advancing the trust region under false pretenses such as a type II error scenario (believing something to be true when in fact it is false such as the accuracy of a model).

IV. Bayesian Hybrid Bridge Function

One of the key issues for multifidelity modeling (VFM) is how to manage the different models of varying fidelity or how to correct the low-fidelity model to approximate the high-fidelity data by making use of so-called bridge functions, which are sometimes called scaling functions. Existing bridge functions can be divided into three categories [15]: additive, multiplicative, and hybrid.

All of these approaches require the construction of an unknown function to correct the lower-fidelity model, which in turn will approximate the high-fidelity model. Note that bridge functions can take the form of low-order polynomials, Kriging models, or any surrogate. It must also be noted that, given the use of a TRMM schema for handling the decision of when to use a given fidelity level, certain criteria must be met by these surrogate models to effectively ensure convergence. The criteria of first-order accuracy is easily addressed through the use of interpolation surrogates.

The key contribution of this paper lies in the development of the Bayesian weighting coefficients. This novel approach allows the use of TRMM techniques with multipoint surrogate models (surrogates built using more than one or two training/construction points). Previous efforts in low-fidelity correction approaches to multifidelity

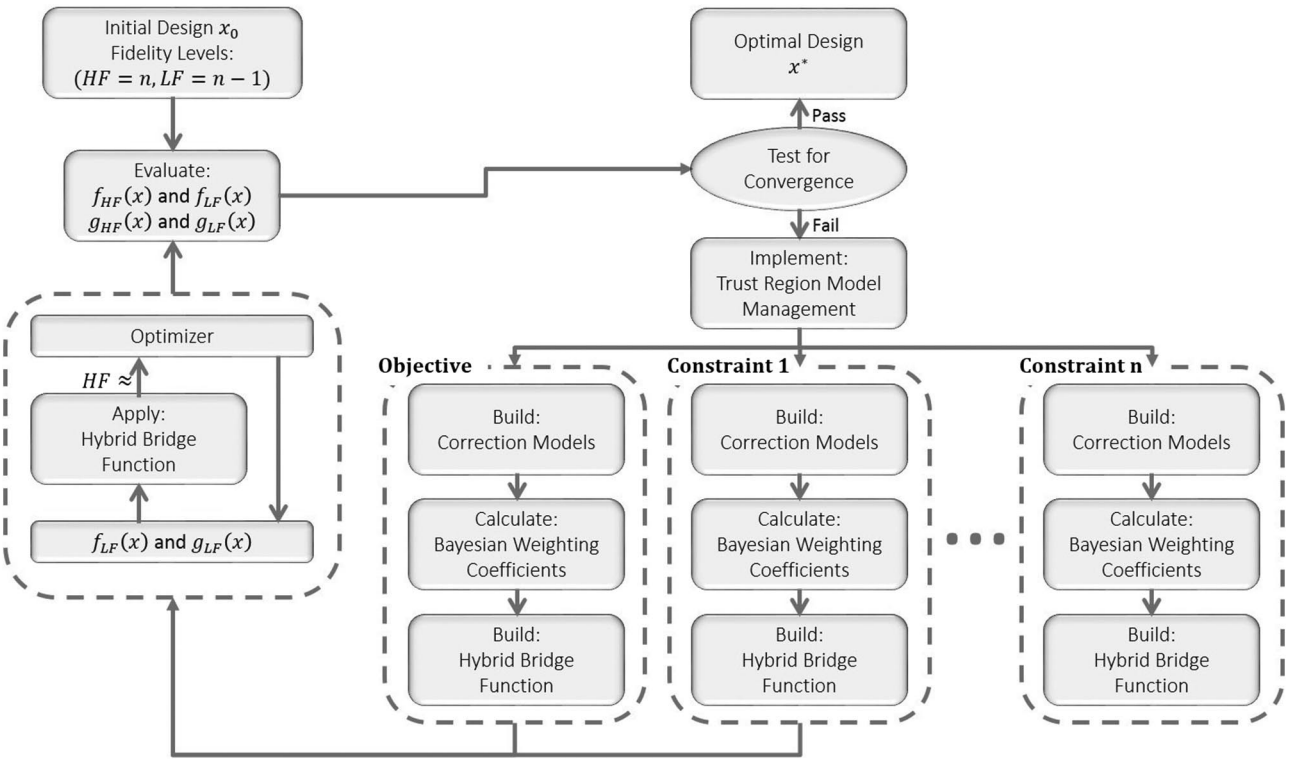


Fig. 2 Multifidelity optimization flow chart.

design require the use of single point approximations due to reliance on the previous optimum point for determining a weighting between additive and multiplicative corrections.

The multiplicative bridge function can be defined through use of

$$\tilde{\phi}_{\text{Mult}}(\boldsymbol{x}) = \tilde{\gamma}(\boldsymbol{x}) f_L(\boldsymbol{x}) \approx f_H(\boldsymbol{x}) \tag{8}$$

In

$$\tilde{\gamma}(\boldsymbol{x}) = \frac{\tilde{f}_H(\boldsymbol{x})}{\tilde{f}_L(\boldsymbol{x})} \tag{9}$$

$\tilde{f}_H(\boldsymbol{x})$ and $\tilde{f}_L(\boldsymbol{x})$ denote the high- and low-fidelity surrogate models, respectively. After the predicted but unknown bridge function $\tilde{\gamma}(\boldsymbol{x})$ has been approximated, the high-fidelity model $f_H(\boldsymbol{x})$ can be approximated by the VFM using Eq. (8), which uses the exact low-fidelity model $f_L(\boldsymbol{x})$.

It can be shown that the function $\tilde{\gamma}(\boldsymbol{x})$ is the scaling ratio between the high-fidelity surrogate and the low-fidelity surrogate, and when it is multiplied by the low-fidelity model, the approximate response of the high-fidelity model is achieved. However, the previously mentioned multiplicative bridge function may cause problems when one of the sampled values of the low-fidelity model is close to zero. In such a case, the values of both the high- and low-fidelity models should be shifted to avoid an error.

To avoid the possible problem of dividing by zero when using multiplicative bridge functions, an additive bridge function was developed. The additive bridge function can be expressed by

$$\tilde{\beta}(\boldsymbol{x}) = \tilde{f}_H(\boldsymbol{x}) - \tilde{f}_L(\boldsymbol{x}) \tag{10}$$

In

$$\tilde{\phi}_{\text{Add}}(\boldsymbol{x}) = f_L(\boldsymbol{x}) + \tilde{\beta}(\boldsymbol{x}) \approx f_H(\boldsymbol{x}) \tag{11}$$

$\tilde{f}_L(\boldsymbol{x})$ and $\tilde{f}_H(\boldsymbol{x})$ denote the low- and high-fidelity models, respectively. After the predicted but unknown bridge function $\tilde{\beta}(\boldsymbol{x})$ has been approximated, the high-fidelity model $f_H(\boldsymbol{x})$ can be approximated by the VFM using Eq. (11) [7].

It can be shown that the function $\tilde{\beta}(\boldsymbol{x})$ is essentially the error or difference between the low and high-fidelity surrogates. When $\tilde{\beta}(\boldsymbol{x})$ is added to the low-fidelity model, the response of the high-fidelity model is obtained.

Gano et al. [15] showed that additive bridge functions are not always better than multiplicative ones. Both have merits and demerits. Hence, Gano et al. developed a hybrid method that combines the multiplicative and additive methods:

$$\tilde{\phi}(\boldsymbol{x}) = w\tilde{\phi}_{\text{Mult}}(\boldsymbol{x}) + (1-w)\tilde{\phi}_{\text{Add}}(\boldsymbol{x}) \tag{12}$$

This allows pure mathematics to make the determination as to whether an additive or multiplicative correction best suits a given problem and implement said correction in a continuous manner.

In Eq. (12), w is a weight coefficient, the determination of w being the key point for the success of this hybrid bridge function. Eldred et al. [4] proposed using the previously evaluated point to adjust the value of w as in

$$w = \frac{f_H(\boldsymbol{x}^{k-1}) - \tilde{\phi}_{\text{Add}}(\boldsymbol{x}^k)}{\tilde{\phi}_{\text{Mult}}(\boldsymbol{x}^k) - \tilde{\phi}_{\text{Add}}(\boldsymbol{x}^k)} \tag{13}$$

In Eq. (13), $f_H(\boldsymbol{x}^{k-1})$ denotes the optimum of the previous optimization step. During the process of optimization, a new value can be computed at each iteration. Updating these weights allows the framework to adapt as the optimization process matures. However, this approach was developed strictly from an optimization standpoint. Here, it is desired to use as much relevant data as possible to encourage the optimization process not only to just find an optimum design but to do so as efficiently and quickly as possible. Thus, it is desirable to use a technique that will determine a weighting between additive and multiplicative corrections utilizing more than a single data point as well as data other than that which was used to construct the correction surrogates. This is essential, given that Kriging surrogates (and all surrogates that satisfy TRMM criteria) are interpolation functions that would result in saying both the additive and multiplicative models are 100% accurate at all build/observation points.

Therefore, a novel method is proposed in this paper for calculating these weighting coefficients. This method follows a Bayesian posterior updating approach that allows for the use of all data available within the weighting region of the current subproblem optimization defined by the previously discussed TRMM methodology, in which the weight region becomes a surrounding region (collocated with upper bounds at a size d larger than the trust region where $1 < d \leq 2$ and lower bounds equivalent to the trust region bounds) of the trust region as shown in Fig. 3. This methodology defines the hybrid bridge function as the form of [16]

$$\tilde{\phi}(\boldsymbol{x}) = w_{\text{Mult}}\tilde{\phi}_{\text{Mult}}(\boldsymbol{x}) + w_{\text{Add}}\tilde{\phi}_{\text{Add}}(\boldsymbol{x}) \tag{14}$$

This approach is intended to improve accuracy as well as mitigate any surrogate-induced inadequacies due to the use of interpolation models. This is resultant from the fact that the weighting coefficients are calculated via data surrounding the trust region, thus choosing the model that is most accurate at trust region bounds. This is important because the use of a TRMM schema typically results in subproblem optima that lie at the respective trust region bounds [17].

In Eq. (14), w_{Mult} and w_{Add} are determined using

$$w^k_{\text{Mult}} = \frac{w^{k-1}_{\text{Mult}}\psi_{\text{Mult}}(\boldsymbol{x})}{w^{k-1}_{\text{Add}}\psi_{\text{Add}}(\boldsymbol{x}) + w^{k-1}_{\text{Mult}}\psi_{\text{Mult}}(\boldsymbol{x})} \tag{15}$$

and

$$w^k_{\text{Add}} = \frac{w^{k-1}_{\text{Add}}\psi_{\text{Add}}(\boldsymbol{x})}{w^{k-1}_{\text{Add}}\psi_{\text{Add}}(\boldsymbol{x}) + w^{k-1}_{\text{Mult}}\psi_{\text{Mult}}(\boldsymbol{x})} \tag{16}$$

It is worth noting that calculation of these weighting coefficients in this manner allows for an adaptive approach to the calculation of said coefficients, given that the current coefficient is dependent upon the previous coefficients in which case the initial weight are defined as $w^0_{\text{Mult}} = w^0_{\text{Add}} = 1/2$ [17]. This approach is essential to the Bayesian-inspired multifidelity optimization (BIMFO) methodology.

In Eqs. (15) and (16), $\psi_{\text{Mult}}(\boldsymbol{x})$ and $\psi_{\text{Add}}(\boldsymbol{x})$ denote the model likelihoods of $\hat{\phi}_{\text{Mult}}(\boldsymbol{x})$ and $\hat{\phi}_{\text{Add}}(t)$, respectively. These model likelihoods are determined using

$$\psi_i(\boldsymbol{x}) = \left(\frac{1}{2\pi\hat{\sigma}^2_{i,\text{MLE}}}\right)^{n/2} \exp\left(-\frac{n}{2}\right) \tag{17}$$

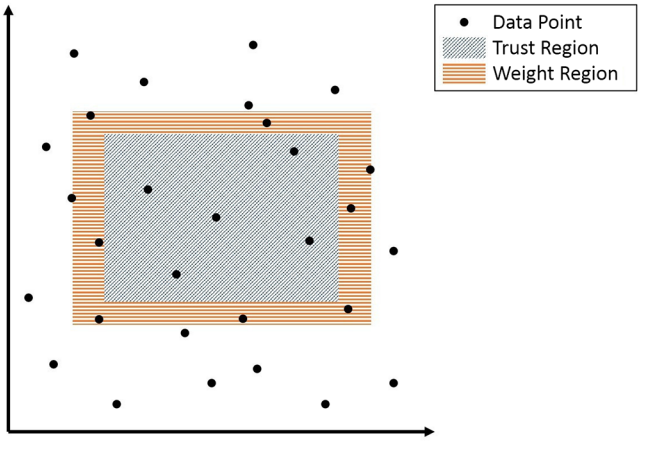


Fig. 3 Weight region collocated with trust region.

and

$$\hat{\sigma}^2_{i,\mathrm{MLE}} = \frac{\sum_{j=1}^{n} \left(f_H(\boldsymbol{x}_j) - \tilde{\phi}_i(\boldsymbol{x}_j)\right)^2}{n} = \mathrm{SSE} \tag{18}$$

where n denotes the number of data points found to be available from previous iterations within the current weight region, i signifies either the multiplicative or additive cases, and j represents the jth data point contained within the current weight region (as discussed earlier and outlined in Algorithm 2).

Note that the model likelihood $\psi_i(\boldsymbol{x})$ is dependent upon the maximum likelihood estimator of model variance $\hat{\sigma}^2_{i,\mathrm{MLE}}$. Therefore, in this formulation, the weighting coefficients are adaptively updated using all available data contained within the kth subproblem optimization weight region.

Use of this Bayesian updating technique allows for the use of multipoint surrogates that have proven to enable faster convergence on complex optimization problems. This technique naturally finds the best correction at/near trust region bounds, which is consequently beneficial. Throughout an optimal design process using trust region/move limit methods, subproblem optima tend to lie near these region bounds, thus choosing the appropriate correction needed to find the optimum. Note that if the optimum does lie within the trust region and this technique does identify a less than perfect correction for finding said optimum the trust region method will identify the corrected low fidelity is inaccurate and redefine an appropriate trust region.

V. Demonstration Problems

The multifidelity optimization procedure presented in this paper is demonstrated on three different computational design problems. These demonstration problems are intended to validate the beneficial application of the proposed multifidelity procedure over traditional design practices. The ability to maximize the use of low-fidelity (computationally cheaper) models to reduce overall computational cost of a design process while maintaining a desired level of accuracy is instrumental in this approach. As is the case, this approach is intended to overcome any scenario in which the low-fidelity model does not satisfactorily capture high-fidelity model trends throughout the entirety of the design space.

This approach to multifidelity design optimization is designed to alleviate the computational cost associated with high-fidelity models. A best-case scenario would result in achieving the same high-fidelity design as achieved via conventional design techniques while reducing the computational cost associated with achieving said design. The actual computational savings expected are very much problem dependent, ranging between 10% and 90% time savings. Therefore, a worst-case scenario would result in still achieving the same high-fidelity design as achieved via conventional design techniques while neither reducing nor increasing the computational time required to achieve said design.

In these demonstrations, the design problems are first solved via conventional design techniques to obtain a baseline design, computational cost, and number of function evaluations to which results of the multifidelity design approach are to be compared. The same design problems are then solved by implementing the proposed Bayesian correction technique. These results are then compared to illustrate the benefits of the proposed techniques. It will be shown that benefits are observed even in a scenario in which the low-fidelity trends oppose high-fidelity trends. The first problem uses the multifidelity optimization design to handle a single set of multiple fidelities in either the objective or constraint but cannot handle both. The second and third problems use the modified approach intended for multiple fidelities in both the objective and the constraints (or multiple constraints). As can be seen in the following results, employing this technique results in converging to the high-fidelity optimum in fewer high-fidelity function evaluations than required to converge to an optimum when performing optimization on just the high-fidelity model.

A. Demonstration Problem 1

The first demonstration problem is a pure academic test case designed to illustrate the proposed multifidelity optimization methodology consisting of the Bayesian hybrid bridge function implemented in a TRMM optimization framework. This problem serves to illustrate the iterative process and convergence of the proposed approach.

1. *Problem Description*

This problem consists of two fidelity levels that can be used to define the objective function of an unconstrained optimal design problem. The high-fidelity objective function is defined as

$$Y_{\mathrm{HF}}(\boldsymbol{x}) = (0.5x - 3)^2 \sin(1.3x + 1) + \exp(-0.25x) \tag{19}$$

whereas the low-fidelity objective is defined by

$$Y_{\mathrm{LF}}(\boldsymbol{x}) = (0.75x - 4)^2 - 7 \tag{20}$$

Both fidelity levels are plotted on the same axes shown in Fig. 4. Note that the high-fidelity objective function is shown as a solid black line, while the solid blue line represents the low-fidelity objective function. As can be seen, the high-fidelity objective has two minima

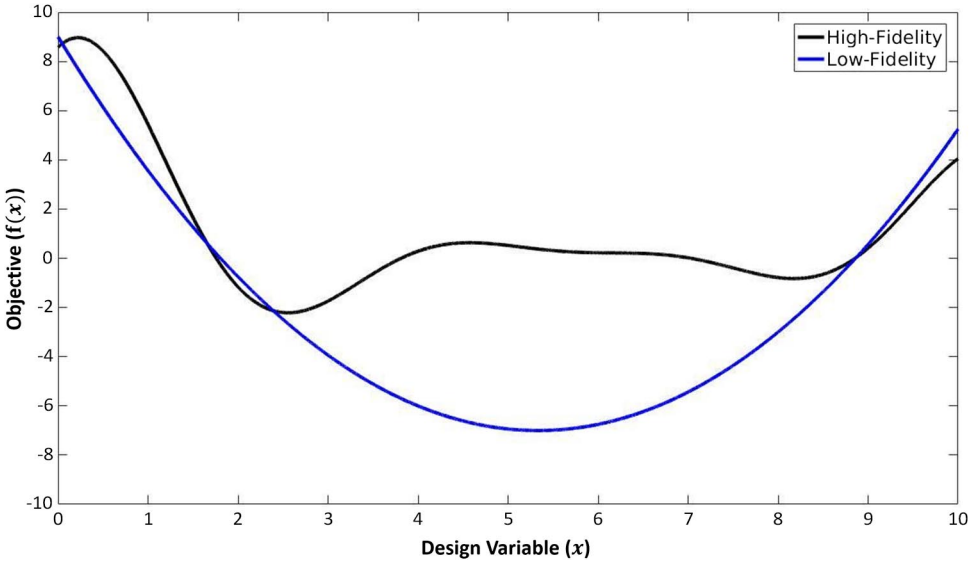


Fig. 4 High- and low-fidelity objective functions.

corresponding to $\boldsymbol{x} = 2.56$ and $\boldsymbol{x} = 8.18$, while the low-fidelity objective has one minima corresponding to $\boldsymbol{x} = 5.34$. It is worth noting that the low-fidelity minima do not correspond to either of the high-fidelity minima and thus cannot lead to a high-fidelity optimum through conventional design techniques.

2. *Multifidelity Optimization Results*

Implementing the proposed multifidelity optimization procedure results in the iterative history depicted in Fig. 5. Note that the trust region in each subfigure is illustrated via a green shaded region in which the center point is represented as a black square and the corresponding iteration optimum is depicted by a red circle. As can be seen, the multifidelity optimization technique converges to an optimum of $x = 2.56$, which directly corresponds to an analytical high-fidelity minima. This optimum design problem shows that even in scenarios in which low- and high-fidelity functions trend in opposing directions, as this problem does over given regions of the design space, the methodology will still converge to a high-fidelity optimum. As can be seen, the trust region size and location adapt as more information becomes available. Likewise, as the trust region adapts, so does the corrected low-fidelity model.

B. Demonstration Problem 2

The second demonstration problem is a pure academic test case designed to ensure the proposed optimization methodology can converge completely to the true optimum. The constraints imposed on this optimization problem can each be defined by one of two equations, each denoting a different fidelity. Likewise, the objective function can be represented by two differing fidelities. Therefore, this problem consists of both a multifidelity objective as well as a multifidelity constraint.

1. *Objective Functions*

Consider the two surfaces shown in Fig. 6, in which Fig. 6a is considered to be the low-fidelity objective (sphere function) represented by

$$f_L(\boldsymbol{x}) = x_1^2 + x_2^2 \tag{21}$$

and Fig. 6b is considered to be the high-fidelity objective (Rosenbrock function) represented by

$$f_H(\boldsymbol{x}) = \left[100(x_2 - x_1^2)^2 + (x_1 - 1)^2\right] \tag{22}$$

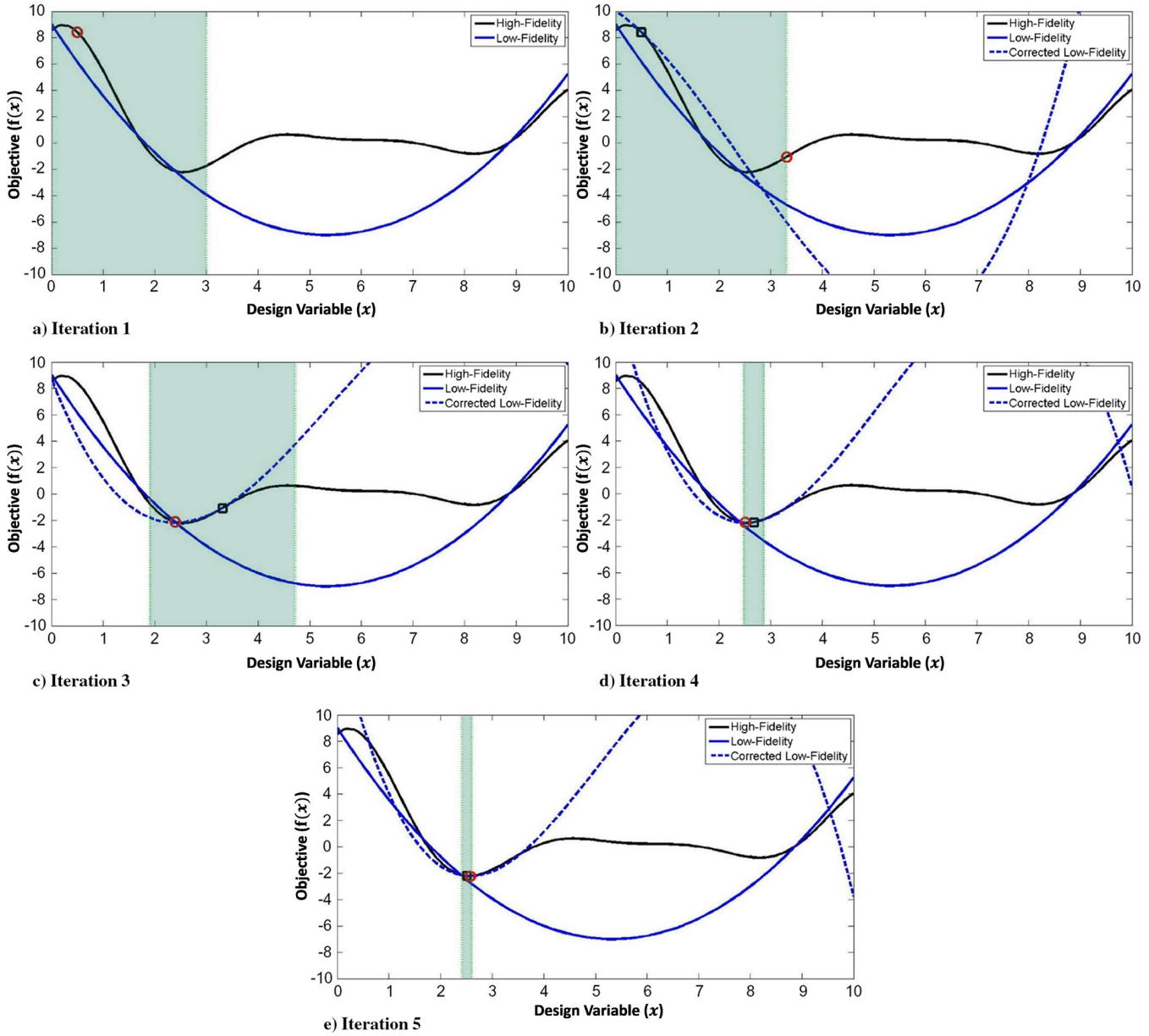
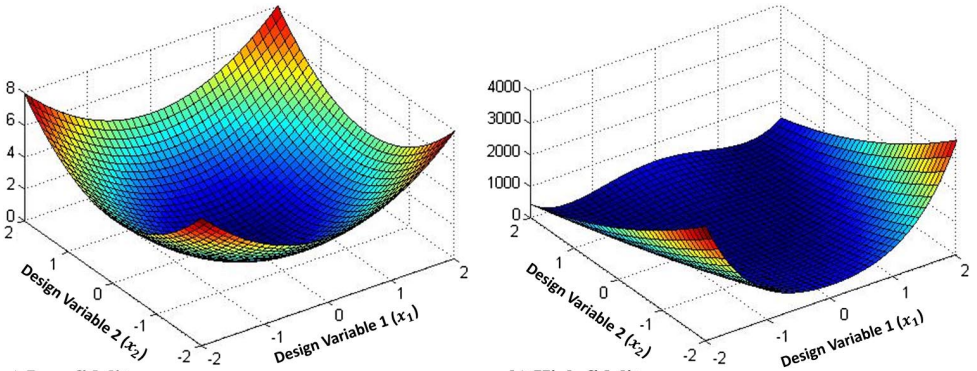


Fig. 5 Demonstration problem 1: iterative history of multifidelity optimization.



a) Low-fidelity b) High-fidelity

Fig. 6 Demonstration problem 2: surfaces described by low- and high-fidelity models.

In this demonstration problem, both models are taken to be two dimensional for the purpose of visualizing the resultant corrected low-fidelity model as well as trust region bounds.

2. *Constraint Functions*

The first constraint representation shown in

$$g_L(\boldsymbol{x}) = x_1 + x_2 - 1.5 \leq 0 \tag{23}$$

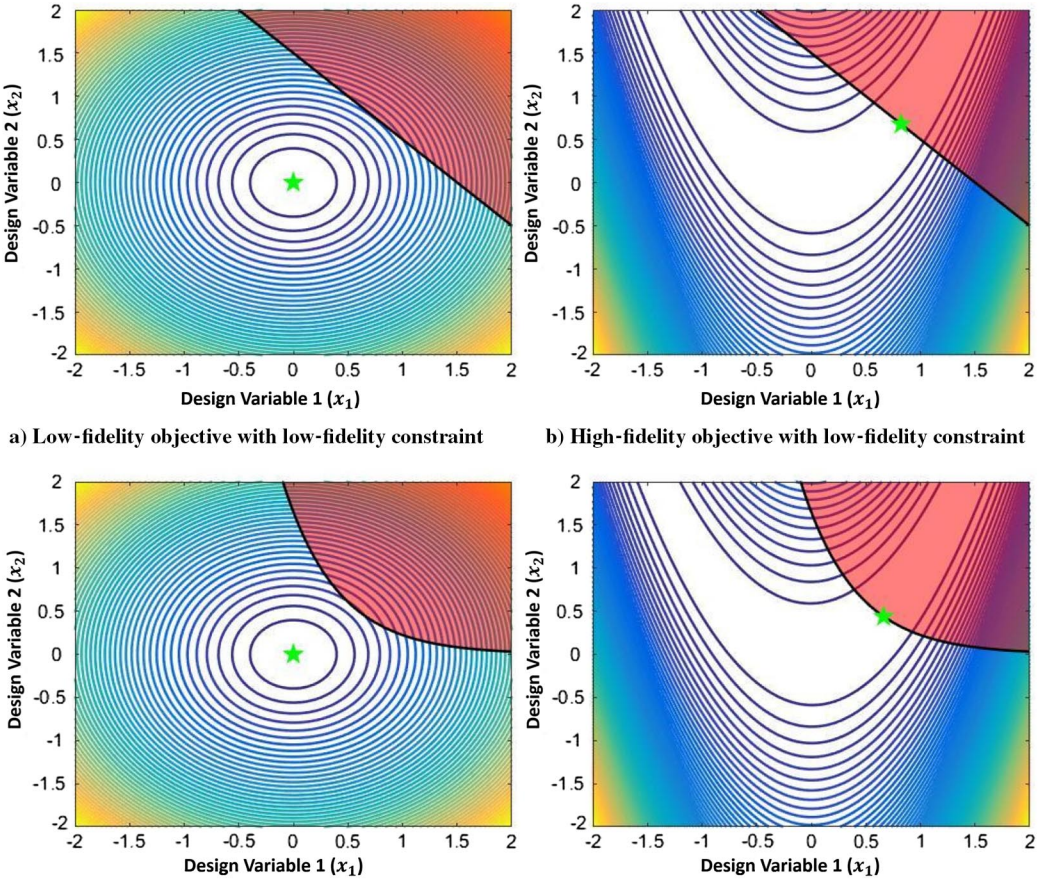
is taken to be low-fidelity, shown in Figs. 7a and 7b over the low- and high-fidelity objectives, respectfully, in which red shading represents an infeasible region. The second constraint representation shown in

$$g_H(\boldsymbol{x}) = x_2 - \exp^{-2x_1+0.5} \leq 0 \tag{24}$$

is taken to be high-fidelity, shown in Figs. 7c and 7d over the low- and high-fidelity objectives, respectfully, in which the red shading represents an infeasible region. In each of these figures, the optimum for the combined objective and constraint illustrated is shown by a green star. As can be observed in Eqs. (23) and (24), the analytical optimum of the high-fidelity objective with high-fidelity constraint is $\boldsymbol{x} = (0.6629, 0.4379)$. It is of interest to note that, given just the low-fidelity objective, neither constraint would be active at the low-fidelity optimum; however, introduction of the high-fidelity optimum activates both low- and high-fidelity constraints, and thus use of strictly low-fidelity in either the objective or constraint would not allow for finding of the true high-fidelity optimum.

3. *Multifidelity Optimization Results*

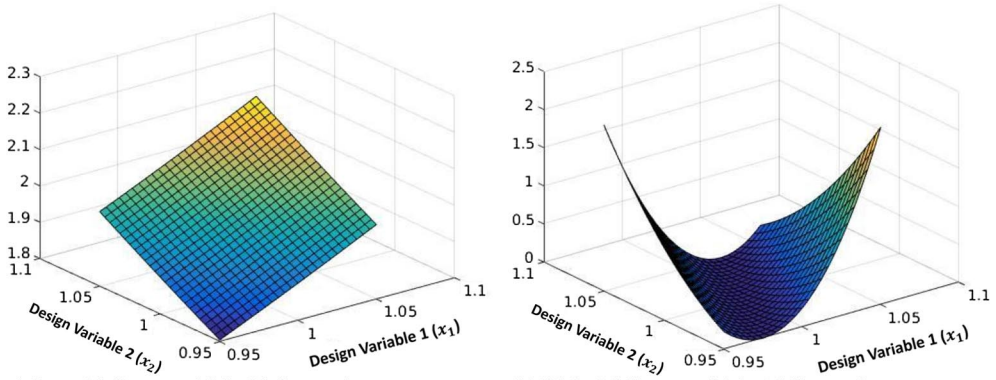
Optimization on the high-fidelity objective and constraint function model arrived at the expected results. This optimization converged to the analytical optimum of $(x_1, x_2) = (0.663, 0.438)$ in 87 high-fidelity function evaluations from an initial point of $(x_1, x_2) = (-2, -2)$. Employing the multifidelity optimization



a) Low-fidelity objective with low-fidelity constraint b) High-fidelity objective with low-fidelity constraint

c) Low-fidelity objective with high-fidelity constraint d) High-fidelity objective with high-fidelity constraint

Fig. 7 Demonstration problem 2 low- and high-fidelity constraints.



a) Low-fidelity near high-fidelity optimum b) High-fidelity near high-fidelity optimum

Fig. 8 Demonstration problem 2: surfaces described by low- and high-fidelity models near high-fidelity optimum.

technique presented in this paper resulted in convergence to an optimum point of $(x_1, x_2) = (0.663, 0.438)$, the analytical optimum, in 21 high-fidelity function evaluations and 137 low-fidelity function evaluations from an initial point of $(x_1, x_2) = (-2, -2)$. However, the key observation is the reduction in the number of high-fidelity function evaluations required to converge to the high-fidelity optimum point. This was a reduction of 66 high-fidelity function evaluations, nearly three quarters the amount required in pure high-fidelity optimization. Note, since these optimum points are not collocated, the models have opposing trends (sensitivities) around the high-fidelity optimum, as can be seen in Figs. 8a and 8b. As noted earlier, some multifidelity optimization approaches such as Cokriging are not capable of handling this type of scenario, and thus converging to a high-fidelity optimum, a scenario in which the low-fidelity sensitivities oppose those of the high-fidelity model. This approach is also unique in the ability to handle multiple fidelities in both the objective and constraint, as is a common scenario in multidisciplinary optimization.

C. Demonstration Problem 3

The third demonstration problem is a two-dimensional aerodynamic airfoil shape optimization. Computational fluid dynamics (CFD) is one specific area in which the need for higher-fidelity physics at a lower computational cost is becoming increasingly necessary in aircraft design. The shape optimization of an airfoil with the objective of minimizing drag coefficient C_{Drag} is subject to a constraint on lift coefficient C_{Lift}. The airfoil shape is parameterized by a Hicks–Henne bump profile imposed upon a NACA 3510 baseline configuration. The shape is controlled via three Hicks–Henne bump parameters as described by

$$y_{\text{top/bot}}(x(j)) = y_{0,\text{top/bot}} + \sum_{i=1}^{n} b^{(i)}_{\text{top/bot}}(x(j)) \tag{25}$$

and

$$b^{(i)}_{\text{top/bot}}(x(j)) = a^{(i)}_{\text{top/bot}} \sin\left(\pi x(j)^{\log 0.5/\log t^{(i)}_{1,\text{top/bot}}}\right)^{t^{(i)}_{2,\text{top/bot}}} \tag{26}$$

In Eq. (25), y_0 references the baseline airfoil (in this case a NACA 3510 airfoil), the subscripts top and bot refer to the top and bottom airfoil surfaces respectively, and $b^{(i)}$ denotes the ith Hicks–Henne bump. Equation (26) defines the Hicks–Henne bump shape used to perturb the airfoil shape as shown in Fig. 9. In this equation, a denotes the magnitude parameter (height of the bump), t_1 is the location parameter (center location of bump), and t_2 is the width parameter (width of bump).

The physics-based models for this demonstration case are constructed using a CFD software package known as FUN3D. FUN3D was developed by NASA as a CFD method that solves the Euler and Navier–Stokes equations in compressible flow. This package can model turbulent and laminar flows in steady or unsteady domains. One of the best components available within FUN3D is the ability to achieve complex step and/or discrete adjoint sensitivities that are leveraged in this work.

1. *Low-Fidelity Model*

The low-fidelity model uses lower-order physics (Euler equations) and a lower quality of mesh. In terms of physics, this model is an Euler-based CFD flow solution assuming inviscid and incompressible flow. The fluid domain is composed of a coarse mesh depicted in Fig. 10a with 2496 computational nodes. This coarse mesh is not desirable for obtaining a fully converged and accurate solution; however, this serves to further demonstrate the ability of the multifidelity optimization methodology given an inferior low-fidelity model. The low-fidelity model is not used for its accuracy but rather its computational efficiency, while still lending physical insight into the flow prediction.

2. *High-Fidelity Model*

The high-fidelity model uses higher-order physics and a higher quality of mesh. In terms of physics, this model is a Reynolds-averaged Navier–Stokes CFD flow solution assuming turbulent and

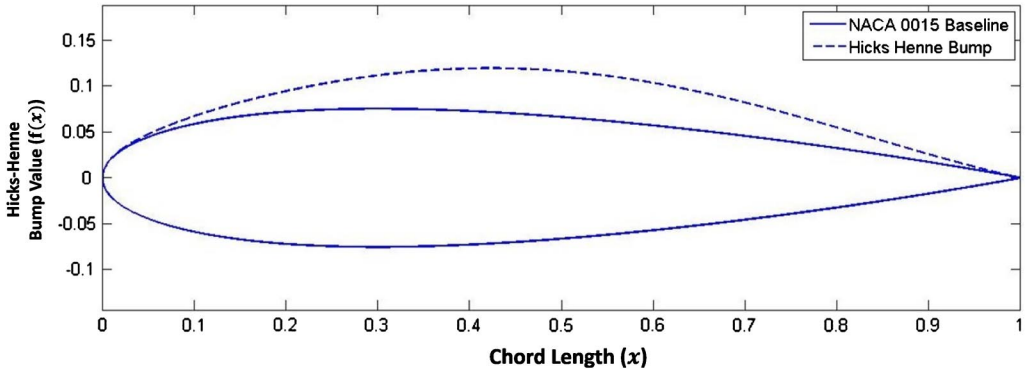
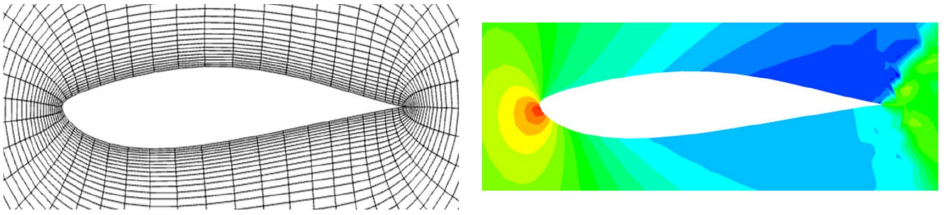
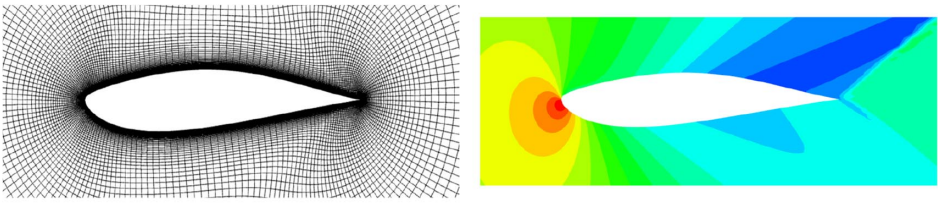


Fig. 9 Hicks–Henne bump parameterization of airfoil shape.



a) Low-fidelity mesh b) Low-fidelity flow pressure

Fig. 10 Low-fidelity FUN3D model.



a) High-fidelity mesh b) High-fidelity flow pressure

Fig. 11 High-fidelity FUN3D model.

compressible flow. The Spalart–Allmaras turbulence model is used to model the turbulent flow. The fluid domain is composed of a fine mesh depicted in Fig. 11a with 39,996 computational nodes. This computational mesh was found to provide converged results in contrast to the low-fidelity mesh.

3. *Flight Conditions*

There are three major flight conditions consistent to both low- and high-fidelity flow solutions. First, the airfoil is immersed in a Mach 1.2 flow. Second, the Reynolds number associated with the flow is equivalent to 6×10^5. Finally, the flow across the airfoil is taken to be at a 5 deg angle of attack.

4. *Optimization Problem*

Shape optimization is to be performed on the described airfoil parameterized by Hicks–Henne bumps. The objective is to minimize drag generated by the airfoil, while maintaining a coefficient of lift equal to or greater than 0.5,

$$\begin{aligned} &\text{Minimize: } f(\boldsymbol{x}) = C_{\text{Drag}} \\ &\text{Subject to: } 0.5 - C_{\text{Lift}} \leq 0 \end{aligned} \tag{27}$$

5. *Multifidelity Optimization Results*

Implementing the proposed multifidelity optimization procedure for a case of multiple fidelities in the objective and constraint converges to an optimum with objective $f(\boldsymbol{x}) = 0.01245$ and constraint $g(\boldsymbol{x}) = 0$ at a design point of $\boldsymbol{x} = (0.015, 0.101, 2.544, 0.062, 0.406, 0.751)$ depicted in Fig. 12. It can be observed that as the amount of high-fidelity information available increases, the accuracy of the corrected low-fidelity model also increases. This can be seen by the reduction in the variation in corrected low-fidelity predictions with an increase in the amount of high-fidelity information. Convergence to optimum was achieved in 49 high-fidelity function evaluations and 531 low-fidelity function evaluations. With a solution time of just over 1 h per high-fidelity function evaluation and just under 1 min per low-fidelity function evaluation. The overall optimization took approximately 57 h on an eight-core hyperthreaded Mac Pro desktop computer.

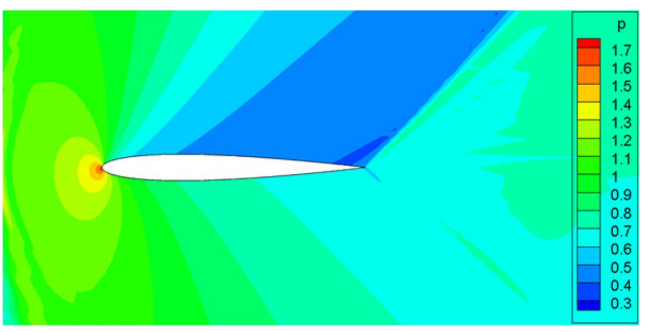


Fig. 12 High-fidelity optimal design point pressure contour.

For comparison, optimization was also performed on the high-fidelity model using a conventional sequential quadratic programming (SQP) optimization technique. This optimization took 189 high-fidelity function evaluations to reach an optimum objective $f(\boldsymbol{x}) = 0.01245$ and constraint $g(\boldsymbol{x}) = 0$ at an optimum design point of $\boldsymbol{x} = (0.015, 0.101, 2.544, 0.062, 0.406, 0.751)$, the exact same optimum design point reached by the multifidelity methodology. These results show that the presented approach converged to a high-fidelity optimum. The computational time required to converge to the high-fidelity optimum took approximately 199 h on the same eight-core hyperthreaded Mac Pro desktop computer. Therefore, implementing this multifidelity optimization methodology reduced computational cost by 71% while still converging to the true (high-fidelity) optimum if even with numerically insignificant and minimal variations.

All multifidelity results have been compared to pure SQP results (high-fidelity-only results) in this paper. Previous work by Fischer and Grandhi [18,19] explored the benefits of an additive approach in comparison to a multiplicative approach as well as hybrid approaches (both Bayesian and standard) comparisons. Previous work also provided a comparison to a pure surrogate-based optimization scenario [19].

VI. Conclusions

In this paper, an adjustment factor technique (surrogate-based Bayesian-influenced hybrid bridge function) combined with a TRMM optimization scheme was presented for use in multifidelity design processes. The novel BHBF discussed is a weighted average of additive and multiplicative adjustment factors in which the weighting coefficients are calculated using a Bayesian updating technique. This technique is the basis of Bayesian statistics and is used in uncertainty quantification and inference-based statistics. The key to the presented surrogate-based adjustment factor technique is the use of a correction surrogate constructed over a localized trust region. This localized trust region is adaptive in the sense that its relative size and location are determined by the accuracy of the corrected low-fidelity model through the presented TRMM methodology. Therefore, a surrogate hybrid bridge function model is constructed using all data available within the trust region (from previous optimization iterations) thus more accurately predicting a high-fidelity response.

This multifidelity optimization methodology was demonstrated using three different optimization problems. The first problem is an

unconstrained minimization in which there are two fidelities that define the objective. This problem proved the capability of arriving at the high-fidelity optimum through the use of a Bayesian-inspired bridge function implemented in a TRMM optimization environment. The second problem extends the first by adding multifidelity constraints to the optimization problem. Such a case is new to literature in handling multiple fidelities in the objective and constraints simultaneously without the use of Lagrange multipliers [9] to simplify the problem via combining constraints in the objective. This problem proved the ability to maintain high-fidelity accuracy while reducing computational cost associated with the design process. Both of these problems also illustrated the ability of the proposed technique to handle opposing trends between fidelity levels. This was shown to be a consequence of using the sphere function as a low-fidelity to with the Rosenbrock function as a high-fidelity given contradictory (opposing) sensitivities (slopes) of the two functions, especially in the region surrounding the Rosenbrock optimum. Finally, the third demonstration illustrated the application and benefits associated with implementing this BHBF correction in a modified TRMM optimization environment on a real-world engineering problem. It was observed that not only did the approach result in a high-fidelity optimum but it did so at 29% of the computational cost required by traditional design optimization techniques.

Sensitivity information is becoming increasingly available with analysis codes. Automatic differentiation techniques as well as the development of recent sensitivity analysis techniques make it relatively simple to generate sensitivity information. Appreciation of the utility of sensitivities in design and optimization have also led to a more widespread implementation of efficient adjoint techniques. With derivatives more readily available, and because sensitivity information is so useful in constructing approximations (as demonstrated in this paper) for the purposes of optimization, it is believed the proposed TRMM scheme with gradient enhanced surrogates of the modified BHBF prove beneficial in multifidelity design. The results obtained using these techniques, and presented in this paper, are promising. Significant computational savings over conventional optimization in terms of high-fidelity simulations have been observed. Despite the sometimes significant dissimilarity between high- and low-fidelity models, this approach was able to capture the descent behavior of the high-fidelity model. When the models are greatly dissimilar, sensitivity-enhanced hybrid bridge functions seem indispensable in obtaining reasonable descent directions, thus leading to efficient optimization. This research is currently focusing on the ability to handle more than two fidelity levels. Current literature only explores bifidelity problems and never addresses how to handle more than two fidelities. There has been promise in addressing these issues through the modification of the optimization scheme. Multifidelity research must address these issues as one commonly has more than two fidelity levels and even multiple models at the same fidelity level. A true multifidelity design approach must handle all aspects present in the multifidelity design problem.

Acknowledgments

The first two authors would like to acknowledge the support provided by the Dayton Area Graduate Studies Institute and the Air Force Research Laboratory through research topic RQ13-8 Development of a Predictive Capability for Multidisciplinary Uncertainty and Sensitivity Analysis. The third author was sponsored by the U.S. Air Force Office of Scientific Research under Laboratory Task 12RB06COR (monitored by Jean-Luc Cambier). This paper has been cleared for public release by the 88th Air Base Wing, Wright–Patterson Air Force Base, of the U.S. Air Force.

References

[1] Allaire, D., Kordonowy, D., Lecerf, M., Mainini, L., and Willcox, K., "Multifidelity DDDAS Methods with Application to a Self-Aware Aerospace Vehicle," *Procedia Computer Science*, Vol. 29, June 2014, pp. 1182–1192. doi:10.1016/j.procs.2014.05.106

[2] Ng, L. W., and Willcox, K. E., "Multifidelity Approaches for Optimization Under Uncertainty," *International Journal for Numerical Methods in Engineering*, Vol. 100, No. 10, 2014, pp. 746–772. doi:10.1002/nme.v100.10

[3] Forrester, A., Sobester, A., and Keane, A., *Engineering Design via Surrogate Modelling: A Practical Guide*, Wiley, Hoboken, NJ, 2008.

[4] Eldred, M., and Dunlavy, D., "Formulations for Surrogate-Based Optimization with Data Fit, Multifidelity, and Reduced-Order Models," *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, AIAA Paper 2006-7117, 2006.

[5] Robinson, T., Eldred, M., Willcox, K., and Haimes, R., "Surrogate-Based Optimization Using Multifidelity Models with Variable Parameterization and Corrected Space Mapping," *AIAA Journal*, Vol. 46, No. 11, 2008, pp. 2814–2822. doi:10.2514/1.36043

[6] Keane, A., "Wing Optimization Using Design of Experiment, Response Surface, and Data Fusion Methods," *Journal of Aircraft*, Vol. 40, No. 4, 2003, pp. 741–750. doi:10.2514/2.3153

[7] Choi, S., Alonso, J. J., and Kroo, I. M., "Two-Level Multifidelity Design Optimization Studies for Supersonic Jets," *Journal of Aircraft*, Vol. 46, No. 3, 2009, pp. 776–790. doi:10.2514/1.34362

[8] Conn, A. R., Scheinberg, K., and Vicente, L. N., "Global Convergence of General Derivative-Free Trust-Region Algorithms to First-and Second-Order Critical Points," *SIAM Journal on Optimization*, Vol. 20, No. 1, 2009, pp. 387–415. doi:10.1137/060673424

[9] Lewis, R. M., and Torczon, V., "A Direct Search Approach to Nonlinear Programming Problems Using an Augmented Lagrangian Method with Explicit Treatment of Linear Constraints," Tech. Rept. WM-CS-2010-01, College of William and Mary, Williamsburg, VA, 2010, pp. 1–25.

[10] Powell, M. J., "The NEWUOA Software for Unconstrained Optimization Without Derivatives," *Large-Scale Nonlinear Optimization*, Springer–Verlag, Berlin, 2006, pp. 255–297.

[11] Kenway, G. K. W., Kennedy, G. J., and Martins, J. R. R. A., "Scalable Parallel Approach for High-Fidelity Steady-State Aeroelastic Analysis and Adjoint Derivative Computations," *AIAA Journal*, Vol. 52, No. 5, March 2014, pp. 935–951. doi:10.2514/1.J052255

[12] Allaire, D., and Willcox, K., "Surrogate Modeling for Uncertainty Assessment with Application to Aviation Environmental System Models," *AIAA Journal*, Vol. 48, No. 8, 2010, pp. 1791–1803. doi:10.2514/1.J050247

[13] Alexandrov, N. M., Lewis, R. M., Gumbert, C. R., Green, L. L., and Newman, P. A., "Approximation and Model Management in Aerodynamic Optimization with Variable-Fidelity Models," *Journal of Aircraft*, Vol. 38, No. 6, 2001, pp. 1093–1101. doi:10.2514/2.2877

[14] Fischer, C. C., and Grandhi, R. V., "Multi-Fidelity Design Optimization via Low-Fidelity Correction Technique," *17th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, AIAA Paper 2016-4293, 2016.

[15] Gano, S. E., Renaud, J. E., and Sanders, B., "Hybrid Variable Fidelity Optimization by Using a Kriging-Based Scaling Function," *AIAA Journal*, Vol. 43, No. 11, 2005, pp. 2422–2433. doi:10.2514/1.12466

[16] Kennedy, M. C., and O'Hagan, A., "Bayesian Calibration of Computer Models," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, Vol. 63, No. 3, 2001, pp. 425–464. doi:10.1111/rssb.2001.63.issue-3

[17] Fischer, C. C., Grandhi, R. V., and Beran, P. S., "Bayesian Low-Fidelity Correction Approach to Multi-Fidelity Aerospace Design," *58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, AIAA Paper 2017-0133, 2017.

[18] Fischer, C. C., and Grandhi, R. V., "Utilizing an Adjustment Factor to Scale Between Multiple Fidelities Within a Design Process: A Stepping Stone to Dialable Fidelity Design," *16th AIAA Non-Deterministic Approaches Conference*, AIAA Paper 2014-1011, 2014.

[19] Fischer, C. C., and Grandhi, R. V., "A Surrogate-Based Adjustment Factor Approach to Multi-Fidelity Design Optimization," *17th AIAA Non-Deterministic Approaches Conference*, AIAA Paper 2015-1375, 2015.

C. Pettit
Associate Editor