

Shawn E. Gano · John E. Renaud · Jay D. Martin · Timothy W. Simpson

# Update strategies for kriging models used in variable fidelity optimization

Received: 25 May 2005 / Revised manuscript received: 21 February 2006 / Published online: 8 July 2006  
© Springer-Verlag 2006

**Abstract** Many optimization methods for simulation-based design rely on the sequential use of metamodels to reduce the associated computational burden. In particular, kriging models are frequently used in variable fidelity optimization. Nevertheless, such methods may become computationally inefficient when solving problems with large numbers of design variables and/or sampled data points due to the expensive process of optimizing the kriging model parameters in each iteration. One solution to this problem would be to replace the kriging models with traditional Taylor series response surface models. Kriging models, however, were shown to provide good approximations of computer simulations that incorporate larger amounts of data, resulting in better global accuracy. In this paper, a metamodel update management scheme (MUMS) is proposed to reduce the cost of using kriging models sequentially by updating the kriging model parameters only when they produce a poor approximation. The scheme uses the trust region ratio (TR-MUMS), which is a ratio that compares the approximation to the true model. Two demonstration problems are used to evaluate the proposed method: an internal combustion engine sizing problem and a control-augmented structural design problem. The results indicate that the TR-MUMS approach is very effective; on the demonstration problems, it reduced the number of likelihood evaluations by three orders of magnitude compared to using a global optimizer to find the kriging parameters in every iteration. It was also found that in trust region-based method, the kriging model parameters need not be updated using a global optimizer—local methods perform just as well in terms of providing a good approximation without affecting

the overall convergence rate, which, in turn, results in a faster execution time.

**Keywords** Kriging models · Metamodels · Fidelity optimization

## 1 Introduction

Simulation-based design methods have recently incorporated metamodels in creative ways to reduce computational expense. Metamodels are derived from an array of sample points evaluated using a simulation model; they can be either interpolative or approximate. Different types of metamodels have been developed for various purposes. Some are locally accurate and computationally efficient to construct and evaluate, making them very attractive for use in sequentially approximate optimization (SAO) methods. Other forms are more global in scope but require larger computational effort to construct and evaluate. However, global metamodels have been successfully used as approximations of the more expensive computer simulations in the design process (Booker 1998; Simpson et al. 2001a). A prominent method to globally approximate deterministic computer simulations is kriging; however, kriging models can be a computational burden when used in a sequential manner (Sacks et al. 1989). The computational expense arises from having to recompute the kriging model parameters after each iteration to include new data points. Methods for reducing the cost of using kriging models sequentially are proposed and evaluated in this research.

Kriging was inspired by the work of Krige, a South African geologist. He proposed innovative concepts for mining estimation but never formalized the method. Then, Georges Matheron developed the Theory of Regionalized Variables (Matheron 1971) based on the work done by Krige and called the method kriging in his honor. A variant of this method, called ordinary kriging, was first applied to deterministic computer simulations by Currin et al. (1988) and called Design and Analysis of Computer Experiments (DACE). Since this time, it has been widely used to approximate computer models (Booker et al. 1999; Gano et al. 2005b; Giunta and Watson 1998; Chung and Alonso 2002b;

Shawn E. Gano · John E. Renaud (✉)  
University of Notre Dame, Notre Dame, IN, 46556-5637, USA  
Tel.: +1-574-6318616  
e-mail: jrenaud@nd.edu

Jay D. Martin  
Applied Research Laboratory, The Pennsylvania State University,  
State College, PA 16804, USA  
e-mail: jdm111@psu.edu

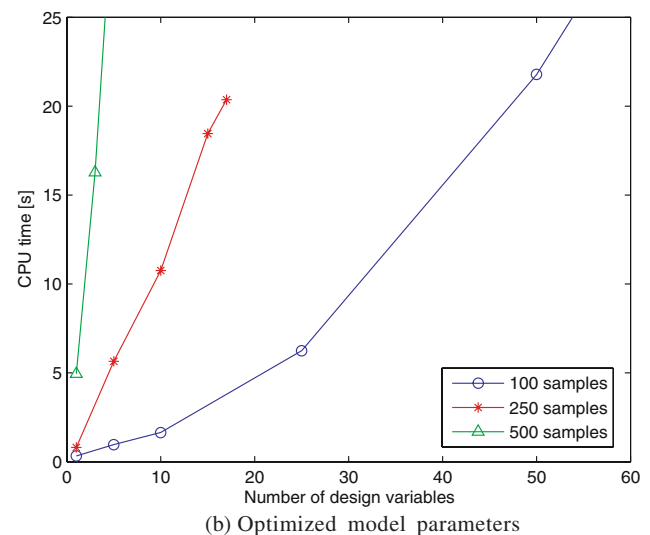
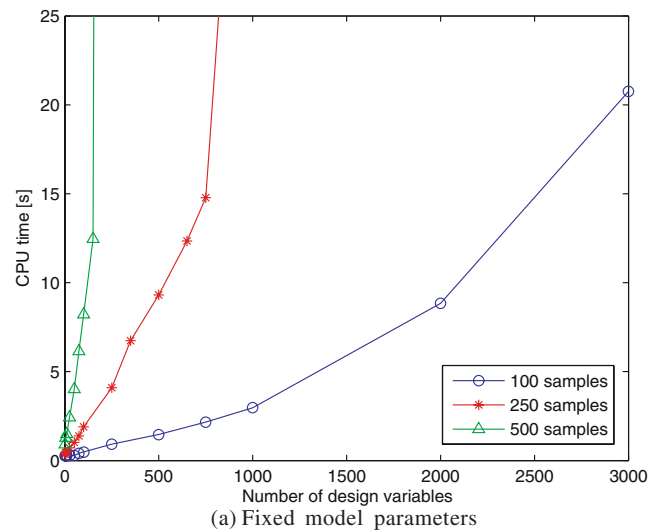
Timothy W. Simpson  
The Pennsylvania State University, University Park, PA 16802, USA  
e-mail: tws8@psu.edu

Simpson et al. 2001c, 2004). Kriging is flexible and quite robust in approximating complex multidimensional functions, making it well-suited for this application.

Kriging methods have been used to model the response of many engineering systems, including the design of a low-boom business jet by Chung and Alonso (2002a) and the design of an aerospike nozzle by Simpson et al. (2001b). Martin and Simpson (2003, 2005) conducted a study on using kriging models to approximate deterministic computer models and discussed the applicability of various kriging variants.

In the optimization field, kriging models have mostly served as surrogate models. However, they were also used in variable fidelity optimization (VFO), which are methods that reduce the cost of finding optima of computationally expensive functions or simulations. Booker et al. (1999), Jones et al. (1998), and Sasena et al. (2002) all used kriging models as surrogates, which were managed through the course of the optimization. Gano et al. (2005b) built kriging-based scaling functions that match low-fidelity models to high-fidelity models, such that the low-fidelity model augmented with the scaling function approximates the high-fidelity model. This scaling approach, combined with a trust region-managed scheme, was proven to converge to the solution of the higher fidelity model with the intent of saving computational resources.

Kriging is typically used as an interpolation process when modeling deterministic computer output. In the kriging formulation, there are a number of model parameters that must be chosen to control what effect nearby sample points have on the prediction of intermediate values. The selection of these parameters can be costly and depends on the dimensionality of the problem and the number of sample points used to construct the kriging model. There are two main methods of estimating these parameters (Martin and Simpson 2005): cross-validation and maximum likelihood estimation (MLE). In this work, only the MLE method is used because it provides an estimate of the variance and the likelihood gives a measure of the model fitness. As its name implies, the MLE method involves an optimization to find the model parameters that maximize the likelihood of the model parameters given the observations of the simulation. The optimization can be performed with either a local optimizer, for example gradient-based or pattern search methods, or using a global stochastic optimization scheme. Ripley (1988), Warnes and Ripley (1987), and Martin and Simpson (2003) discuss problems with this optimization. They state two main difficulties: (1) the function is prone to have multiple maxima and (2) the function may have long, almost flat ridges of near optimal values. To overcome these difficulties, Martin and Simpson (2004) suggested the use of simulated annealing, which is a stochastic optimization method. Simulated annealing, however, is much more computationally expensive than a gradient-based method. This paper addresses the question: is the extra expense to build a possibly better kriging model worthwhile in terms of the upper level optimization scheme (i.e., variable fidelity methods) in which it is used?



**Fig. 1** Relative cost of building kriging models with different numbers of samples and design variables. **a** Fixed model parameters. **b** Optimized model parameters

Because of the sequential nature of the variable fidelity optimization process, the kriging models are rebuilt to include new information in each iteration. This process of rebuilding the kriging model by refitting the model parameters can be quite costly, as shown in Fig. 1. The figure shows the cost trend of constructing a kriging model with increasing numbers of samples and design variables; the CPU time on the vertical axis should be used as a relative scale due to the wide variation in computer processing speeds. Figure 1a shows the trends for constructing a kriging model with fixed model parameters, while Fig. 1b accounts for optimizing the model parameters using a pattern search optimizer. For large problems with many design variables or samples, the cost of rebuilding the kriging model could reduce or outweigh the savings of using such methods. When the kriging models are rebuilt, a maximum likelihood estimator is used to find *optimal* parameter values. However, the interpolative nature

of kriging does not depend on the model parameters used; therefore, the kriging models may not need to be rebuilt, i.e., parameters refit, each time a new data point is added. In this paper, a metamodel update management scheme (MUMS) is proposed to determine when the kriging model parameters should be updated; the method is based on using the trust region ratio value (TR-MUMS). This research also determines if such a management scheme can predict whether the kriging model parameters, when updated, should be found using a local optimization update or with a more expensive but more robust stochastic method.

In this paper, a description of basic kriging theory is presented first (Section 2) along with the local and global optimization approaches used to fit the kriging model parameters. Then a brief overview of the variable fidelity method used to test the kriging model update strategies is given (Section 3). Next, a description of the update schemes is provided (Section 4), as well as the numerical procedure for their evaluation (Section 5). The update strategies are then demonstrated on an internal combustion engine design problem and a control-augmented structural design problem (Section 6). Finally, conclusions and future work are given (Section 7).

## 2 Ordinary kriging using maximum likelihood estimation

A kriging model can be built using all of the points in which computer simulations were evaluated. Kriging models are used in this research to scale a low-fidelity model to match a higher fidelity model in the variable fidelity optimization framework in Section 3 using all past high- and low-fidelity simulation evaluations. A brief overview of the methodology for constructing a kriging model is given in this section.

Kriging begins by estimating an unknown function,  $y$ , with the form (Denison et al. 2002; Stein 1999; Jones et al. 1998)

$$\hat{y}(\mathbf{x}) = B + Z(\mathbf{x}). \quad (1)$$

The term  $B$  is an unknown constant or global trend function, typically linear or quadratic.  $Z(x)$  is one realization of sample path of a Gaussian process with a mean of zero, a variance  $\sigma^2$ , and a covariance of

$$\text{Cov}[Z(x^i), Z(x^j)] = \sigma^2 \mathbf{R}[R(\mathbf{x}^i, \mathbf{x}^j)], \quad (2)$$

where  $\mathbf{R}$  is the correlation matrix,  $R$  is the correlation function selected by the user, and  $i$  and  $j$  run from 1 to  $n_s$ . It is important to notice that  $\mathbf{R}$  is symmetric and has unit values along the diagonal.

The selection of the correlation function is chosen by the user when generating the kriging model. In the statistical and engineering literature (Simpson et al. 2001b; Giunta and Watson 1998), the Gaussian function is, by far, the most popular and is also used in this work. It is defined as

$$R(x^i, x^j) = e^{-\sum_{k=1}^{n_v} \theta_k |x_k^i - x_k^j|^2}, \quad (3)$$

where  $\theta_k$  is the vector of unknown correlation parameters which is of length  $n_v$ , the number of design variables. Also,  $x_k^i$  and  $x_k^j$  are the  $k$ th elements of the sample points  $\mathbf{x}^i$  and  $\mathbf{x}^j$ .

The kriging model estimates values of  $y(\mathbf{x})$  at untried values of  $\mathbf{x}$  given set of expected values or sample points. The mean squared error is defined as the square of the expected difference between the real and the approximated response at any point. Mathematically, this is stated as

$$MSE = E(y(\mathbf{x}) - \hat{y}(\mathbf{x}))^2. \quad (4)$$

Because kriging is an interpolation process, the model will have no  $MSE$  at a sample point. If the  $MSE$  is minimized, then the kriging predictor is

$$\hat{y} = \mathbf{f}\hat{B} + \mathbf{r}^T(\mathbf{x})\mathbf{R}^{-1}(\mathbf{y} - \mathbf{f}\hat{B}), \quad (5)$$

where  $\mathbf{y}$  is the vector of responses (objective values) to the sample locations  $\{\mathbf{x}^1, \dots, \mathbf{x}^{n_s}\}$  and  $\mathbf{f}$  is a constant vector of all ones with length  $n_s$ . In (5), the correlation vector,  $\mathbf{r}(\mathbf{x})$ , is the correlation between the value at a new location  $\mathbf{x}$  and the values at the sampled locations. To use this predictor,  $\hat{B}$  and  $\theta_k$  must both be found, as  $\mathbf{r}$  and  $\mathbf{R}$  depend on  $\theta_k$ . The correlation vector is

$$\mathbf{r}(\mathbf{x})^T = [R(\mathbf{x}, \mathbf{x}^1), \dots, R(\mathbf{x}, \mathbf{x}^{n_s})]. \quad (6)$$

The unknown parameters  $\theta_k$  are found using maximum likelihood estimation (Giunta and Watson 1998). This approach uses the likelihood of the assumed Gaussian computer model with kriging parameters of  $\theta_k$  given observations  $\mathbf{y}$ . This likelihood is defined as

$$L(\theta_k|\mathbf{y}) = \frac{e^{-\frac{(\mathbf{y}-\mathbf{f}\hat{B})^T \mathbf{R}(\theta_k)^{-1} (\mathbf{y}-\mathbf{f}\hat{B})}{2\hat{\sigma}^2}}}{\sqrt{(2\pi\hat{\sigma}^2)^{n_s} |\mathbf{R}(\theta_k)|}}. \quad (7)$$

It is more common to use the log of the likelihood for mathematical convenience, which simplifies (7) to

$$l(\theta_k|\mathbf{y}) = \frac{-(\mathbf{y} - \mathbf{f}\hat{B})^T \mathbf{R}(\theta_k)^{-1} (\mathbf{y} - \mathbf{f}\hat{B})}{2\hat{\sigma}^2} - \frac{n_s \ln(2\pi\hat{\sigma}^2) + \ln(|\mathbf{R}(\theta_k)|)}{2}. \quad (8)$$

By setting the derivatives of (8) with respect to both  $\hat{B}$  and  $\hat{\sigma}^2$  to zero, a closed-form solution for the optimal values of the trend function,  $\hat{B}$ , and estimated variance,  $\hat{\sigma}^2$ , are found:

$$\hat{B} = (\mathbf{f}^T \mathbf{R}^{-1} \mathbf{f})^{-1} \mathbf{f}^T \mathbf{R}^{-1} \mathbf{y}, \quad (9)$$

$$\hat{\sigma}^2 = \frac{1}{n_s} \left( (\mathbf{y} - \mathbf{f}\hat{B})^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{f}\hat{B}) \right). \quad (10)$$

Finally, the correlation parameters,  $\theta_k$ , are found by maximizing the log-likelihood function, known as the maximum likelihood estimation (MLE) problem, which can be reduced to:

$$\begin{aligned} &\text{Maximize}_{\theta_k} && -\frac{n_s \ln(\hat{\sigma}^2) + \ln(|\mathbf{R}|)}{2} \\ &\text{subject to:} && 0 < \theta_k \leq \infty. \end{aligned} \quad (11)$$

This is a  $n_v$  dimensional optimization problem, which is well-posed. Notice that a kriging model can be built for any values of  $\theta_k$ ; this optimization ensures the *best choice* for the  $\theta_k$  values. In practice, when solving for  $\theta_k$ ,  $\mathbf{R}$  may become badly scaled; this is overcome using various numerical techniques as described by Lophaven et al. (2002a,b).

The MLE problem in (11) is difficult to solve because of the multimodal and long near-optimal ridge properties of the likelihood function. To address these issues, stochastic optimizers were used. However, such methods, while more robust in finding the global solution to the MLE problem, are also much more expensive in terms of function calls. In this research, three different optimizers are used: Quasi-Newton, which is a gradient based local optimizer; pattern search (Kowalik and Osborne 1968), which is a local optimizer that doesn't use gradient information; and adaptive simulated annealing (ASA; Ingber 1989, 1992, 1993, 1996), which is a global optimization approach. The trade-off between global and local solutions and function call expense is studied in this research of sequentially using kriging models. A one-dimensional example comparing each of these optimizers to solve the MLE problem can be found in Gano et al. (2005a).

### 3 Variable fidelity optimization

The typical framework for variable fidelity optimization is depicted in Fig. 2 and is based, in part, on work done by Alexandrov et al. (2001) and is fully described by Gano et al.

(2005b). This framework is designed to reduce the number of high-fidelity function calls during the optimization process by using a scaling function and lower fidelity models.

The variable fidelity optimization problem can be written using the standard nonlinear programming problem:

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimize}} && f(\mathbf{x}) \\ & \text{subject to:} && \mathbf{g}(\mathbf{x}) \leq 0 \\ & && \mathbf{h}(\mathbf{x}) = 0 \\ & && \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}, \end{aligned}$$

where  $f$  is the objective function,  $\mathbf{x}$  is the vector of design variables,  $\mathbf{g}$  is the vector of inequality constraints,  $\mathbf{h}$  is the vector of equality constraints, and  $\mathbf{l}$  and  $\mathbf{u}$  are the upper and lower bounds, respectively, of the design variables. The objective and the constraints can be evaluated using various fidelity models, and the solution is desired of the highest fidelity model.

The following process describes the basic steps of the variable fidelity optimization framework:

**Step 1 Initialization**

The objective and constraints are evaluated using both the high- and low-fidelity models at the starting design point,  $\mathbf{x}_0$ . Also, an initial  $l_1$  penalty function is evaluated (see step 5).

**Step 2 Gradient evaluation**

The gradient of the objective and the Jacobian for the constraints are evaluated using both the high- and low-fidelity models at the current design point,  $\mathbf{x}_n$ .

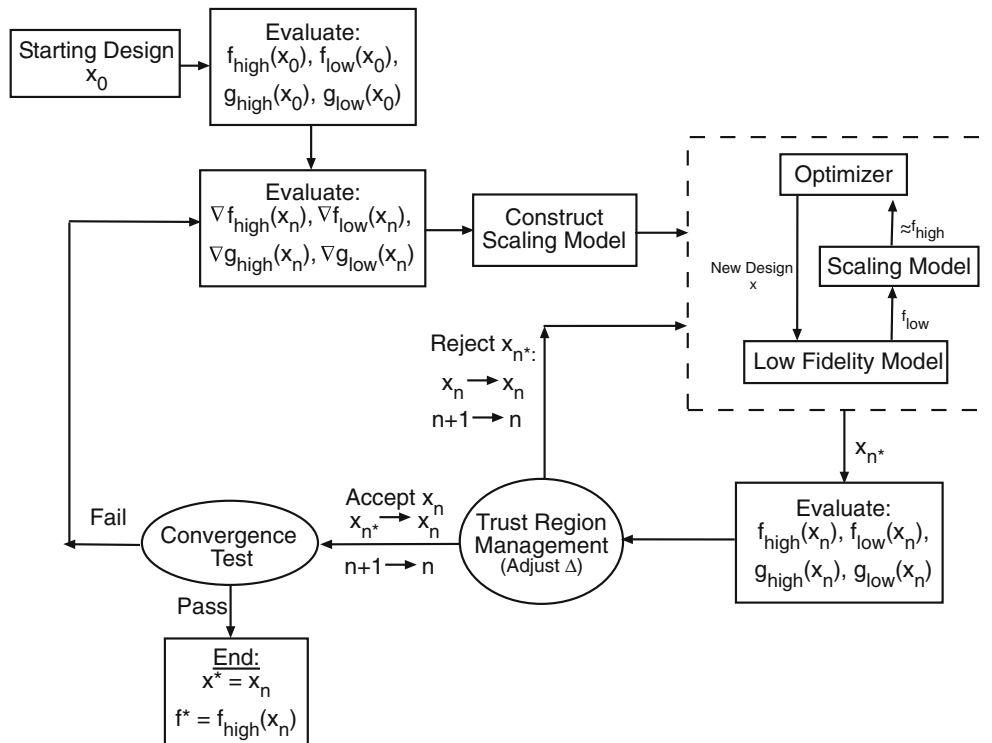


Fig. 2 Variable fidelity framework flowchart

**Step 3 Construct scaling model**

A scaling model is constructed to insure matching between the fidelity models. This model can be based on many different methods; additive and multiplicative are the most common and are discussed in more detail later. Each method can be modeled as first-order, second-order, or kriging-based. A scaling model is constructed for each constraint, as well as for the objective function. The scaling models are applied directly to the low-fidelity objective and constraints.

**Step 4 Optimize-scaled low-fidelity model**

The low-fidelity model scaled with the scaling model constructed in step 3 is optimized. The choice of optimizer used is based on preference. In the work done by Alexandrov et al. (2000a), three optimizers were compared: augmented Lagrangian method, multilevel algorithms for large-scale constrained optimization (Alexandrov et al. 2000b; used for coupled MDO problems), and sequential quadratic programming (SQP). For typical single discipline problems, Alexandrov found SQP to be the most promising, and it is used in this research. The unscaled constraints are included in this step to ensure that they are always satisfied.

**Step 5 Evaluate new design and  $l_1$  penalty function**

Using the resulting design point from step 4, the high-fidelity objective and constraints are evaluated. The objective and constraint values are used to calculate a current value of the  $l_1$  penalty function,  $P$ , for the high-fidelity and scaled low-fidelity models. The penalty function is defined as

$$P(\mathbf{x}) = f(\mathbf{x}) + \frac{1}{\mu_n} \sum \max(0, g_i(\mathbf{x})) + \frac{1}{\mu_n} \sum |h_i(\mathbf{x})|, \quad (12)$$

where  $\mu$  is the penalty weight which is typically decreased by a factor of ten each time a new point is accepted. This penalty weighting drives all the active constraints to zero as the algorithm converges.

**Step 6 Trust region management**

To guarantee convergence of the variable fidelity optimization framework, a trust region model management strategy is employed (Conn et al. 1988). This method provides a means for adaptively managing the allowable move limits for the approximate design space. Originally, these methods were used to ensure the convergence of Newton-based methods.

A trust region ratio allows the trust region model management framework to monitor how well the approximation matches the high-fidelity design space. After each completed optimization on the scaled low-fidelity model, a new candidate point,  $\mathbf{x}_n^*$ , is found. A trust region ratio,  $\rho_n$ , is calculated at this new point:

$$\rho_n = \frac{P(\mathbf{x}_n)_{high} - P(\mathbf{x}_n^*)_{high}}{P(\mathbf{x}_n)_{scaled} - P(\mathbf{x}_n^*)_{scaled}}, \quad (13)$$

where  $P()_{high}$  and  $P()_{scaled}$  are the  $l_1$  penalty functions for the high-fidelity and scaled low-fidelity models and the point  $\mathbf{x}_n$ , which was the initial point of the optimization. The value of  $\rho_n$  is the ratio of the actual change in the function to the predicted change of the function by the scaled lower fidelity model. Because the constraints are also approximated, the trust region ratio must account for this and converge to a feasible design, which is why the  $l_1$  penalty function is used. The trust region size is governed by the following standard rules (Giunta and Eldred 2000; Rodriguez et al. 2001):

$$\Delta_{n+1} = \begin{cases} c_1 \Delta_n : \rho_n \leq R_1 \vee \rho_n > R_3 \\ \Delta_n : R_1 < \rho_n < R_2 \\ \Gamma \Delta_n : R_2 \leq \rho_n \leq R_3 \end{cases}. \quad (14)$$

where  $\Gamma = c_2$  if  $\|x_n^* - x_n\|_\infty = \Delta_n$ , otherwise  $\Gamma = 1$ . A typical set of values for the range limiting constants are  $R_1 = 0.25$ ,  $R_2 = 0.75$ , and  $R_3 = 1.25$ , while the trust region multiplication factors are typically  $c_1 = 0.25$  and  $c_2 = 3$ . Physically,  $\rho$  represents how good of an approximation our scaled low-fidelity model is compared to the high-fidelity model. If  $\rho$  is near 1, the approximation is quite good. If  $\rho$  is near zero, then the approximation is not as good, but it still captures the minimization trend. If  $\rho$  is negative, then the point is a worse design. In this case the point is rejected, the trust region size is reduced by the factor  $c_1$ , and the algorithm returns to step 4. As long as  $\rho > 0$ , the point is accepted and the algorithm proceeds to step 7.

**Step 7 Convergence test**

For the implementation used in this research, convergence was determined by the following stopping criterion:

$$f_{high}(\mathbf{x}_n) - f_{high}(\mathbf{x}_{n-1}) < \epsilon_f, \quad (15)$$

$$\|\mathbf{x}_n - \mathbf{x}_{n-1}\| < \epsilon_x, \quad (16)$$

where  $\epsilon_f$  and  $\epsilon_x$  are tolerances supplied by the user, and  $n$  is the current iteration counter. If any of the two inequalities at the current point is true, the algorithm is considered converged. If the convergence test is true, then the final design is found, otherwise, the algorithm returns to step 2.

Convergence to a Karush–Kuhn–Tucker (KKT) point can be tested at the final design point by evaluating the projected gradient, as done by Rodríguez and Renaud (1998). If the projected gradient at the final design is zero, then the design satisfies the necessary first-order KKT conditions; if the projected Hessian is positive definite, then the point is indeed a KKT point.

**3.1 Scaling methods**

Existing variable fidelity or approximate model management frameworks come in two varieties: multiplicative or additive.

Currently, the most common is the multiplicative framework, devised by Alexandrov and Lewis (2003) based on Haftka's (Haftka 1991; Chang et al. 1993) scaling function. The additive method was presented by Lewis and Nash (2000). Both methods are based on constructing an unknown function to update the lower fidelity model, which in turn will approximate the higher fidelity model. In this paper the additive method is used, but similar results are obtained using the multiplicative method.

In the additive method, a set of high- and low-fidelity models,  $f_{high}(\mathbf{x})$  and  $f_{low}(\mathbf{x})$ , are matched by adding the low-fidelity model to an unknown function  $\gamma(\mathbf{x})$ . This is expressed mathematically as

$$f_{high}(\mathbf{x}) = f_{low}(\mathbf{x}) + \gamma(\mathbf{x}). \quad (17)$$

The additive scaling function can be solved for by subtracting the low-fidelity function from both sides:

$$\gamma(\mathbf{x}) = f_{high}(\mathbf{x}) - f_{low}(\mathbf{x}). \quad (18)$$

From (18), it is clear that the function  $\gamma(\mathbf{x})$  is the additive scaling of the high-fidelity model to the low-fidelity model or the error between them. When this function is added to the low-fidelity model, the response of the high-fidelity model is produced. Because the additive function is not known, it is typically approximated using first- or second-order Taylor series models. A similar function for the constraints can be developed in the same manner as (17) and (18).

### 3.2 Kriging-based scaling models for variable fidelity optimization

The scaling models developed in the first- and second-order approaches are only local to the current design point and do not use past information. When using variable fidelity physics-based models, the low-fidelity model typically is a global model. Therefore, a global scaling function may be better at approximating the high-fidelity response, especially scaling approaches that do not rely on assuming any underlying functional form of the model as in the case of the first- and second-order approaches. In this investigation, a kriging-based scaling function is used to improve the scaling between the different fidelity models on a more global scale (Gano et al. 2005b). This approach allows the use of all information calculated throughout the course of the optimization, even when the trust region ratio is less than zero and the corresponding design point is rejected. The kriging model can be constructed for any type of scaling function, for example, the additive or multiplicative methods already discussed.

The kriging model gives exact responses at sample points, as it is an interpolating function. This ensures that at least zero-order matching is obtained. With the inclusion of gradient information using co-kriging (Chung and Alonso 2002a), first-order matching is achieved in this research. First-order matching combined with the trust region model management strategy typically provides for a provably convergent framework (Alexandrov et al. 2001). However, because of the use of kriging models as scaling functions in this work, the proof

of convergence may not hold theoretically, but in practice, the method performs well as seen in the results and previous research (Gano et al. 2005b).

Another benefit of using a kriging-based scaling approach is that past data can be easily incorporated into the scaling model to further increase the convergence rate. Often, a model is evaluated for various purposes before an optimization is performed. These results can be included in the kriging model to improve its matching capabilities.

## 4 Metamodel update management strategy for sequentially building kriging models

This section discusses an update strategy that can be used to reduce the cost of having to optimize the kriging model parameters at each iteration of a sequential approximation optimization (SAO) method-like variable fidelity optimization. The strategy uses an indicator to try to measure how accurate a kriging model with a given set of correlation parameters is and if the parameters should be updated or not after new samples are included. The method will be referred henceforth as TR-MUMS; this naming convention is used because the method uses a trust region ratio value, which gives a measure of how well the approximation matches the true model. Another method called L-MUMS, which used ratios of consecutive kriging model likelihood values, was introduced and found to be not as effective by Gano et al. (2005a).

The TR-MUMS approach to managing the updating of the kriging model parameters uses the trust region ratio  $\rho_n$ , defined in (13). This ratio is already calculated in variable fidelity optimization as it is in most SAO methods. The trust region ratio provides a measure of how well the approximation represents the true model. Therefore, it can be used to estimate when the kriging model parameters need to be updated—when the kriging model is producing a poor approximation. Unlike the likelihood ratio described by Gano et al. (2005a), this method only works after a bad approximation is produced and does nothing to determine if a kriging model is poor before it is used. However, this approach is more physically intuitive and simpler. The update scheme is defined as:

$$K_{n+1} = \begin{cases} \text{update} & : \rho_n \leq A_1 \\ \text{noupdate} & : \rho_n > A_1 \end{cases}. \quad (19)$$

The parameter  $A_1$  does not necessarily have to correspond to the  $R_1$  value used in updating the trust region size in (14). This update scheme can use a global optimizer or a local optimizer when the ratio is sufficiently far from unity and skips the update of the kriging parameters when the ratio is near one.

## 5 Numerical experimental procedure and demonstration problems

This section describes the numerical procedure used for evaluating the kriging model updating strategies. The goal is to

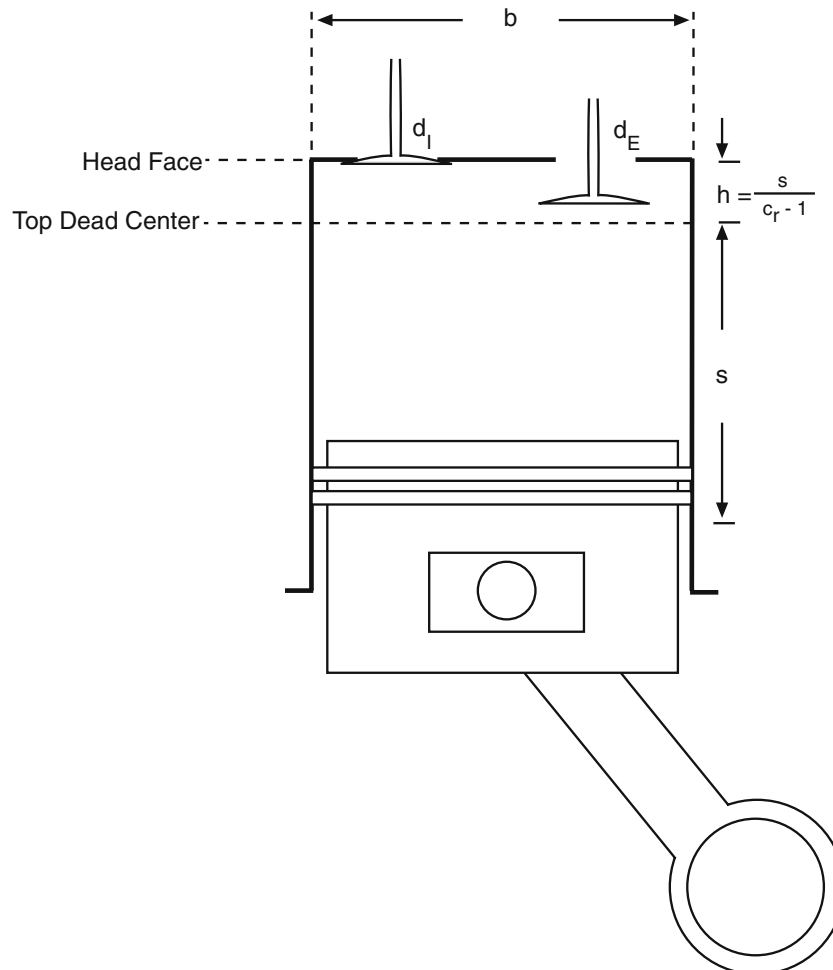
determine if such updating strategies can reduce the cost of building the kriging models over the course of the optimization process without unduly increasing the number of iterations and high-fidelity function calls required for convergence. Two engineering design problems were used to study the kriging update strategies: (1) an internal combustion engine sizing problem and (2) a control-augmented structural design problem.

Three main steps were used to evaluate the management of updating the kriging model parameters. First, the test problems were optimized using the variable fidelity method, in which the kriging model parameters were updated at each iteration; this was repeated using the various likelihood optimization methods. Second, the variable fidelity framework was run using the same kriging model parameters for the entire optimization process. The fixed values for these parameters corresponded to using each of the likelihood optimizers after the first iteration. These results allow for a comparison between always and never updating the kriging model and show how this affects the convergence of the variable fidelity algorithm. Finally, the different MUMS were used and compared to the results from the first two studies.

The two demonstration problems are described in the next two sections. Quadratic response surfaces (Myers and Montgomery 1995) were used as the low-fidelity objectives and constraints for both problems. They were generated using a high-fidelity Latin hypercube sampling (Iman et al. 1981a,b) using the number of design variables squared data points. The computational costs of evaluating the high- and low-fidelity models for the engine design problem were comparable and are used solely to evaluate the savings of the kriging update schemes. For the control-augmented structure problem, the high-fidelity model required 150 times the computational cost of the low-fidelity model to evaluate.

### 5.1 Internal combustion engine design

In this problem, the geometry for a flat head internal combustion chamber is sought to provide maximal specific power. The design must also satisfy a number of constraints including packaging, fuel economy, and knock limitations. The problem was originally posed by the Ford Motor Corporation (Wagner and Papalambros 1991), and a robust variation



**Fig. 3** Combustion chamber geometry

**Table 1** Starting and optimum designs for the internal combustion engine design problem

Design variable	Starting design	Optimum design
$b$ (mm)	75	83.33
$c_r$ (L/L)	6.42	9.45
$d_E$ (mm)	26	30.99
$d_I$ (mm)	39	37.34
$w$ (rpm)	7,500	6,070
$f$ (kW/L)	30.28	55.67

of the problem was solved by McAllister and Simpson (2003). The engine analysis parameters were determined by Ford's engine assessment model. A schematic for the flat head combustion chamber is shown in Fig. 3. The design variables for this problem are the cylinder bore  $b$ , the compression ratio  $c_r$ , exhaust valve diameter  $d_E$ , intake valve diameter  $d_I$ , and the revolutions per minute at peak power  $w$ . The design problem is mathematically posed below.

$$\text{minimize } f = K_0 \left( \frac{\rho Q}{A_f} \eta_t \eta_v - FMEP \right) w \quad (20)$$

subject to

$$K_1 N_c b - L_1 \leq 0, \quad (21)$$

$$\sqrt{\frac{4K_2 V}{\pi N_c L_2}} - b \leq 0, \quad (22)$$

$$d_I + d_E - K_3 b \leq 0, \quad (23)$$

$$K_4 d_I - d_E \leq 0, \quad (24)$$

$$d_E - K_5 d_I \leq 0, \quad (25)$$

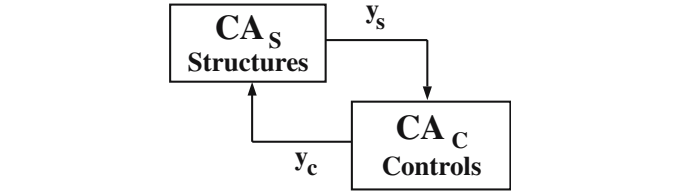
$$9.428 \times 10^{-5} \frac{4V}{\pi N_c} \frac{w}{d_I^2} - K_6 C_s \leq 0, \quad (26)$$

$$c_r - 13.2 + 0.045b \leq 0, \quad (27)$$

$$w - K_7 \leq 0, \quad (28)$$

$$3.6 \times 10^6 - K_8 Q \eta_{tw} \leq 0. \quad (29)$$

The thermal  $\eta_t$ , volumetric  $\eta_v$ , and thermal at part load point efficiencies  $\eta_{tw}$  are all functions of the design variables and are given in the original paper (Wagner and

**Fig. 5** Dependency diagram of the control-augmented structure design problem

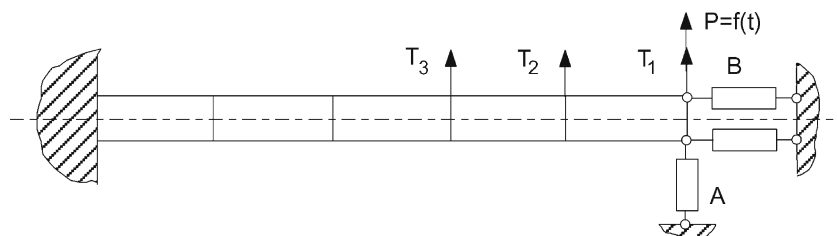
Papalambros 1991). The original paper also includes expressions for the friction mean effective pressure (FMEP), density of the inlet charge  $\rho$ , lower heating value  $Q$ , air-to-fuel ratio  $A_f$ , number of cylinders  $N_c$ , displacement volume  $V$ , port discharge coefficient  $C_s$ , and the parameters  $K_i$ ,  $i = \{0 \dots 12\}$  and  $L_i$ ,  $i = \{1, 2\}$ . The starting and optimal designs are given in Table 1 along with their respective objective function values.

## 5.2 Control-augmented structure design

The control-augmented structure design problem, shown in Fig. 4, was introduced by Sobieszczanski-Sobieski et al. (1991). The problem comprises a total of 11 design variables and 43 states. The physical problem consists of a cantilever beam subjected to static loads along the beam and to a dynamic excitation force applied at the free end. Two sets of actuators are placed at the free end of the beam to control both the lateral and rotational displacement.

The system analysis is comprised of two coupled contributing analyses, as shown in Fig. 5. The structures subsystem,  $CA_s$ , consists of a finite element model of the beam where the natural frequencies and modes of the cantilever beam are computed.  $CA_s$  requires, in addition to the characteristics of the beam, the weight of the control system as input. The weight of the control system is calculated at the controls  $CA_c$ . The weight of the control system is a function of the dynamic displacements and rotations of the free end of the beam. These dynamic displacements and rotations are functions of the natural frequencies and modes obtained in the structures  $CA$ , thus subjecting these  $CA$ s to coupling.

The objective of the optimization is to minimize the total weight of the system  $W_t$ , composed of the weight of the beam  $W_s$  plus the weight of the control system  $W_c$ . The de-

**Fig. 4** Cantilever beam with actuators

**Table 2** Starting and optimum designs for the control-augmented structure problem

Design variable	Starting design	Optimum design
$b_1$ (in)	10.0	3.0
$b_2$ (in)	10.0	3.0
$b_3$ (in)	10.0	3.0
$b_4$ (in)	10.0	3.0
$b_5$ (in)	10.0	3.0
$h_1$ (in)	10.0	13.85
$h_2$ (in)	10.0	11.96
$h_3$ (in)	10.0	9.78
$h_4$ (in)	10.0	7.06
$h_5$ (in)	10.0	3.75
$c$	0.01	0.06

sign variables consist of heights  $h_i$  and widths  $b_i$  of each of the five beam elements, and a control gain parameter  $c$ . The minimization is subjected to seven constraints on the static stresses, lateral and rotational displacements, natural frequencies, and dynamic lateral and rotational displacements at the free end of the beam. The problem is posed as:

$$\begin{aligned}
 &\text{minimize } W_t = W_s + W_c \\
 &\text{subject to } 1 - \frac{dl}{dl_a} \geq 0, \\
 &\quad 1 - \frac{dr}{dr_a} \geq 0, \\
 &\quad \frac{\omega_1}{\omega_{1a}} - 1 \geq 0, \\
 &\quad \frac{\omega_2}{\omega_{2a}} - 1 \geq 0, \\
 &\quad 1 - \frac{\sigma}{\sigma_a} \geq 0, \\
 &\quad 1 - \frac{ddl}{ddl_a} \geq 0, \\
 &\quad 1 - \frac{ddr}{ddr_a} \geq 0,
 \end{aligned}$$

**Table 3** Internal combustion engine optimization results

Optimizer	Iteration	High function calls	Low function calls	L evals	$\theta$ updates
$\theta$ updated every iteration					
ASA	7	42	911	344,307	7
Quasi-Newton	6	31	308	8,668	6
Pattern search	7	32	453	407	7
TR-MUMS ( $\theta$ updated when $\rho_n < 0.25$ )					
ASA	7	42	698	56,701	1
Quasi-Newton	5	25	274	2,852	2
Pattern search	5	25	462	118	2
$\theta$ updated once					
ASA	7	42	698	56,701	1
Quasi-Newton	6	31	247	347	1
Pattern search	5	25	364	62	1
Standard single fidelity optimization					
SQP	9	64	–	–	–

where  $dl$  is the static lateral displacement,  $dr$  is the static rotational displacement,  $ddl$  is the dynamic lateral displacement,  $ddr$  is the dynamic rotational displacement,  $\omega_1$  is the first natural frequency,  $\omega_2$  is the second natural frequency, and  $\sigma$  is the static stress. The subscript  $a$  stands for the allowed value. The optimum for this problem is presented in Table 2. The minimum weight  $W = 1493.6 \text{ lbs}$  occurs where 6 design variables are at their bounds.

## 6 Results

The results of optimizing the internal combustion engine and control-augmented structural problems using the numerical procedure outlined in the previous section to evaluate the effectiveness of the TR-MUMS strategy are given and discussed in this section.

### 6.1 Internal combustion engine design results

A summary of all of the results of optimizing the combustion chamber of an internal engine is given in Table 3. The number of iterations, high-fidelity function calls, low-fidelity function calls, likelihood evaluations, and the number of times the kriging model parameters were updated are all summarized in the table. All derivatives were evaluated using finite differencing, and each of these evaluations is reflected in the high- and low-fidelity function evaluation counts.

The first section of Table 3 shows the results for the variable fidelity optimization when the kriging model parameters were updated in each iteration. For comparison, the number of function calls required to optimize the problem using a standard SQP solver is also given at the bottom of the table, and for each case, the variable fidelity optimization requires fewer high-fidelity function calls. The results show that updating the kriging models using ASA actually resulted in a higher number of high-fidelity function calls. This suggests that the highest likelihood value may not always produce the best approximation. The number of low-fidelity function calls was also higher using ASA. The largest difference between

the results from updating the kriging models for each iteration is the number of likelihood evaluations required; ASA required two orders of magnitude more evaluations than did the Quasi-Newton optimizer, which in turn required another order of magnitude more evaluations than did the pattern search approach.

The next study repeated the same problem setup but used the value of the kriging model parameters found in the first iteration for the entire optimization process. These results are given in the third section of Table 3. As expected, the number of likelihood evaluations for each method significantly decreased, but still retained the same relative scale between the different optimizers. Unexpectedly, these results are not worse than when the kriging models were updated after each iteration. In some cases, the results are even better. This provides evidence that sequential optimization techniques such as VFO may have low sensitivity to the values used in computing the kriging model and may not need to be updated as frequently. The pattern search method actually converged faster than in the previous trial, while all three methods required fewer low-fidelity function calls. The unexpected result may be related to the theoretical findings of Lim et al. (1997); they noted the kriging best linear unbiased predictor has special asymptotic properties when the output of a computer model is highly correlated over the design space.

In the last trial, the TR-MUMS was used. In this strategy, the kriging model was updated using different optimizers when the trust region ratio was below 0.25 ( $\rho_n < 0.25$ ). The results are given in the second section of Table 3. The results show a decrease in the number of high-fidelity function calls using Quasi-Newton and the pattern search approaches as compared to the initial study. The pattern search used the same number of high-fidelity function calls as in the second trial. Both of these methods updated the kriging model parameters one extra time as compared to using the same kriging models for the entire optimization, allowing for the improved performance. As expected, the number of likelihood function evaluations increased accordingly, but was still much less than updating the parameters after each iteration. These results show that the TR-MUMS can decrease the cost of variable fidelity optimization.

## 6.2 Control-augmented structure design results

The computational expense of the optimization process using the different kriging model update schemes is more closely studied using the control-augmented structures problem. The amount of computational time required, wall time, was added to the summary of results for this problem, which is given in Table 4. A high-fidelity function call required 1.5 s to evaluate, the low-fidelity model took 0.01 s to evaluate, and to optimize the kriging models, on average, took 3 s and 900 s for the pattern search and ASA approaches, respectively. Also, the Quasi-Newton optimizer was not used in this problem for finding the kriging models. This was done without loss of generality because it is a local optimizer as is the pattern search, and both had similar results in the previous demonstration problem relative to ASA.

The first section of Table 4 shows the results when the kriging model parameters were updated in each iteration. As in the internal combustion problem, all derivatives were evaluated using finite differencing, and the function evaluations are reflected in the total function counts. The ASA approach used slightly fewer high-fidelity function calls than did the pattern search and it also used fewer low-fidelity function calls as well. The main difference between the two optimizers was in the number of likelihood evaluations required. The ASA method used over a million likelihood evaluations, three orders of magnitude more than the pattern search method. This huge difference of likelihood evaluations is the reason the execution time for using ASA was much higher, even higher than performing standard optimization on the high-fidelity function alone. The ASA approach could have been less computationally expensive than the standard SQP optimization if the computational cost of the high-fidelity model was much greater, as it would be in some engineering design problems.

The third section of Table 4 shows the results from using a single set of values for computing the kriging approximation throughout the optimization process. In this case, the variable fidelity optimization process failed to converge to the optimal solution for both methods. The premature convergence was due to the fact that the trust region size became

**Table 4** Control-augmented structure optimization results

Optimizer	Iteration	High function calls	Low function calls	L evals	$\theta$ updates	Wall time (s)
$\theta$ updated every iteration						
ASA	12	123	1,573	1,070,000	12	11,013
Pattern search	16	127	2,093	1,594	16	259
TR-MUMS ( $\theta$ updated when $\rho_n < 0.25$ )						
ASA	14	125	1,894	410,009	5	4,706
Pattern search	17	128	2,149	839	8	237
$\theta$ updated once						
ASA <sup>a</sup>	17	117	1,430	100,016	1	1,090
Pattern search <sup>a</sup>	35	333	4,214	140	1	545
Standard single fidelity optimization						
SQP	26	340	–	–	–	510

<sup>a</sup>These did not converge to the optimum; the trust region size became too small

too small, which is an indication that the kriging approximation became unusable and needed to be updated. This result is contrary to the results found in the internal engine design problem, showing that there is a need to update the kriging model parameters during the VFO process.

The TR-MUMS scheme for updating the kriging model parameters converged to the optimal solution using fewer than half of the number of kriging model updates than updating the model parameters in every iteration. The results are given in the second section of Table 4. The reduction in likelihood evaluations, however, came at a cost of slightly increasing the number of high- and low-fidelity function calls, at least for this problem. These results indicate that, for problems in which the cost of updating the kriging model parameters is significantly high relative to a high-fidelity function call, the trust region ratio can be used as a good indicator as to when the kriging parameters should be updated.

## 7 Conclusions

Optimizing the kriging model parameters can be computationally expensive, especially when the number of design variables and/or sample points used are large. This cost could outweigh the benefits of using kriging models as approximations in trust region-managed, sequential approximate optimization methods such as variable fidelity optimization, which attempt to decrease the computational cost of simulation-based design. In this paper, an overview of the variable fidelity method was given along with basic kriging theory. Then, different ways of optimizing the kriging model parameters were compared. The likelihood function, which is the objective in determining the kriging model parameters, can be multimodal and have long flat ridges, making it tough for local optimization methods such as Quasi-Newton or pattern search to converge to the global optima. Global optimization approaches, like adaptive simulated annealing, were used by researchers to address this problem. However, such methods required many orders of magnitude more likelihood evaluations to converge. A metamodel update management scheme, TR-MUMS, was proposed in this paper to reduce the cost of using kriging models that are sequentially updated. TR-MUMS used the trust region ratio value to determine when the kriging model was not doing an adequate job of approximating the true model and, therefore, needed to be updated.

Two engineering design problems were solved to study the sensitivity of the variable fidelity optimization framework to different kriging model parameter updating schemes. The test problems included an internal engine combustion chamber sizing problem and a control-augmented structural design problem. It was found that the variable fidelity method was insensitive to what optimizer was used; in fact, the simulated annealing optimizer required many more likelihood evaluations and did not improve performance. Local optimization methods like Quasi-Newton and, especially, the pattern search method performed well with many fewer likelihood

evaluations. The trust region ratio was found to be a good indicator of when to update the kriging mode parameters, and its use in the TR-MUMS strategy resulted in fewer kriging model parameter updates and a lower total cost of optimization. Further tests using TR-MUMS on a broader spectrum of test problems should be done before any strong conclusions can be made; however, this work indicates that this strategy can significantly reduce the computational time in using kriging models in variable fidelity methods.

Future work on metamodel update management schemes could include using an improved trend function in the kriging model as recommended by Martin and Simpson (2005). Using a nonconstant trend function will address the issues of multimodality and long correlation ranges. The trend function itself will be able to better model the long correlations, allowing the correlation function to model the shorter correlations, resulting in reduced multimodality of the likelihood function. Using a nonconstant trend function also results in faster optimization of the MLE problem when using gradient methods.

**Acknowledgements** This research effort was supported in part by the following grants and contracts: AFRL/DARPA/Anteon Corporation Contract F33615-98-D-3210, ONR Grant N00014-02-1-0786, and NSF Grant DMI-0114975.

## References

- Alexandrov NM, Lewis RM (2003) Large-scale PDE-constrained optimization. In: Biegler LT, Ghattas O, Heinkenschloss M, Bloemen Waanders Bv (eds) Lecture notes in computational science and engineering, vol. 30. Springer, Berlin Heidelberg New York
- Alexandrov NM, Lewis RM, Gumbert CR, Green LL, Newman PA (2000a) Optimization with variable-fidelity models applied to wing design. In: Proceedings of the 38th AIAA Aerospace Sciences Meeting and Exhibit, No. AIAA 2000-0841, Reno, Nevada, 2000
- Alexandrov NM, Nielsen EJ, Lewis RM, Anderson WK (2000b) First-order model management with variable-fidelity physics applied to multi-element airfoil optimization. In: 8th AIAA/NASA/USAF/ISSMO Symposium on multidisciplinary analysis and optimization, No. AIAA paper 2000-4886, Long Beach, CA, 2000
- Alexandrov NM, Dennis JE, Lewis RM, Torczon V (2001) A trust region framework for managing the use of approximation models in optimization. ICASE Report 97-50, NASA/CR-201745, October 2001
- Booker AJ (1998) Design and analysis of computer experiments. In: Proceedings of the 7th AIAA/USAF/NASA ISSMO Symposium on multidisciplinary analysis and optimization, vol 1. St. Louis, MO, 1998, pp 118–128
- Booker AJ, Dennis JE, Frank PD, Serafini DB, Torczon V, Trosset MW (1999) A rigorous framework for optimization of expensive function by surrogates. *Struct Optim* 17(1):1–13
- Chang KJ, Haftka RT, Giles GL, Kao P-J (1993) Sensitivity-based scaling for approximating structural response. *J Aircr* 30(2):283–288
- Chung H-S, Alonso JJ (2002a) Using gradients to construct cokriging approximation models for high-dimensional design optimization problems. In: 40th AIAA aerospace sciences meeting and exhibit, No. AIAA 2002-0317, Reno, NV, 2002
- Chung H-S, Alonso JJ (2002b) Design of a low-boom supersonic business jet using cokriging approximation models. In: 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, No. AIAA 2002-5598, Atlanta, GA, 2002

- Conn AR, Gould NIM, Toint PL (1988) Global convergence of a class of trust region algorithms for optimization with simple bounds. *SIAM J Numer Anal* 25(2):433–464
- Curran C, Mitchell T, Morris M, Ylvisaker D (1988) A Bayesian approach to the design and analysis of computer experiments. Technical Report ORNL-6498, Oak Ridge National Laboratory, Oak Ridge, TN
- Denison DGT, Holmes CC, Mallick BK, Smith AFM (2002) Bayesian methods for nonlinear classification and regression. Wiley, New York
- Gano SE, Renaud JE, Martin JD, Simpson TW (2005a) Update strategies for kriging models for use in variable fidelity optimization. In: Proceedings of the 1st multidisciplinary design optimization specialist conference, No. AIAA 2005-2057, Austin, TX, 2005
- Gano SE, Renaud JE, Sanders B (2005b) Hybrid variable fidelity optimization using a kriging-based scaling function. *AIAA J* 43(11):2422–2430
- Giunta AA, Watson LT (1998) A comparison of approximation modeling techniques: polynomial versus interpolating models. In: Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on multidisciplinary analysis and optimization, No. AIAA-98-4758, St. Louis, MO, 1998, pp. 392–404
- Giunta AA, Eldred MS (2000) Implementation of a trust region model management strategy in the DAKOTA optimization toolkit. In: Proceedings of the 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, No. AIAA-2000-4935, Long Beach, CA
- Haftka RT (1991) Combining global and local approximations. *AIAA J* 29(9):1523–1525
- Iman RL, Helton JC, Campbell JE (1981a) An approach to sensitivity analysis of computer models, Part I: introduction, input variable selection and preliminary variable assessment. *J Qual Technol* 13:174–183
- Iman RL, Helton JC, Campbell JE (1981b) An approach to sensitivity analysis of computer models, Part II: ranking of input variables, response surface validation, distribution effect and technique synopsis. *J Qual Technol* 13:232–240
- Ingber L (1989) Very fast simulated re-annealing. *J Math Comput Model* 12:967–973
- Ingber L (1992) Genetic algorithms and very fast simulated reannealing: a comparison. *J Math Comput Model* 16(11):87–100
- Ingber L (1993) Simulated annealing: practice versus theory. *J Math Comput Model* 18(11):29–57
- Ingber L (1996) Adaptive simulated annealing (ASA): lessons learned. *J Control Cybern* 25(1):33–54
- Jones DR, Schonlau M, Welch WJ (1998) Efficient global optimization of expensive black-box functions. *J Glob Optim* 13(4):455–492
- Kowalik J, Osborne MR (1968) Methods for unconstrained optimization problems. American Elsevier, New York
- Lewis RM, Nash SG (2000) A multigrid approach to the optimization of systems governed by differential equations In: 8th AIAA/NASA/USAF/ISSMO Symposium on multidisciplinary analysis and optimization, No. AIAA2000-4890, Long Beach, CA, 6–8 September 2000
- Lim YB, Sacks JJSW, Welch WJ (1997) Design and analysis of computer experiments when the output is highly correlated over the input space. Technical report 62, National Institute of Statistical Sciences, NC, USA
- Lophaven SN, Nielsen HB, Søndergaard J (2002a) DACE a Matlab kriging toolbox. Technical Report IMM-REP-2002-12, Technical University of Denmark, version 2.0
- Lophaven SN, Nielsen HB, Søndergaard J (2002b) Aspects of the Matlab toolbox DACE. Technical report IMM-REP-2002-13, Technical University of Denmark
- Martin JD, Simpson TW (2003) A study on the use of kriging models to approximate deterministic computer models. In: Proceedings of ASME 2003 design engineering technical conferences and computers and information in engineering conference, No. DET2003/DAC-48762, Chicago, IL, 2003
- Martin JD, Simpson TW (2004) A Monte Carlo simulation of the kriging model. In: Proceedings of the 11th AIAA/ISSMO multidisciplinary analysis & optimization conference, No. AIAA 2004-4483, Albany, NY, 30 September 2004
- Martin JD, Simpson TW (2005) On the use of kriging models to approximate deterministic computer models. *AIAA J* 43(4):853–863
- Matheron G (1971) The theory of regionalized variables and its applications. Ecole des Mines, Fontainebleau
- McAllister CD, Simpson TW (2003) Multidisciplinary robust design optimization of an internal combustion engine. *ASME J Mech Des* 125:124–130
- Myers RH, Montgomery DC (1995) Response surface methodology: process and product optimization using designed experiments. Wiley, New York
- Ripley BD (1988) Statistical inference for spatial processes. Cambridge University Press, Cambridge, UK
- Rodríguez JF, Renaud JE (1998) Convergence of trust region augmented Lagrangian methods using variable fidelity approximation data. *Struct Optim* 15:141–156
- Rodríguez JF, Pérez VM, Padmanabhan D, Renaud JE (2001) Sequential approximate optimization using variable fidelity response surface approximations. *Struct Multidisc Optim* 22(1):24–34
- Sacks J, Welch WJ, Mitchell TJ, Wynn HP (1989) Design and analysis of computer experiments. *Stat Sci* 4(4):409–423
- Sasena MJ, Papalambros P, Goovaerts P (2002) Exploration of metamodeling sampling criteria for constrained global optimization. *Eng Optim* 34(3):263–278
- Simpson TW, Maurey TM, Korte JJ, Mistree F (2001a) Kriging metamodels for global approximation in simulation-based design. *Eng Comput* 17(2):129–150
- Simpson TW, Maurey TM, Korte JJ, Mistree F (2001b) Kriging metamodels for global approximation in simulation-based multidisciplinary design optimization. *AIAA J* 39(12):2233–2241
- Simpson TW, Peplinski JD, Koch PN, Allen JK (2001c) Metamodels for computer-based engineering design: survey and recommendations. *Eng Comput* 17:129–150
- Simpson TW, Booker AJ, Gosh D, Gunita AA (2004) Approximation methods in multidisciplinary analysis and optimization: a panel discussion. *Struct Multidisc Optim* 27(5):302–313
- Sobieszcanski-Sobieski J, Bloebaum CL, Hajela P (1991) Sensitivity of control-augmented structure obtained by a system decomposition method. *AIAA J* 29(2):264–270
- Stein ML (1999) Interpolation of spatial data: some theory for kriging. Springer, Berlin Heidelberg New York
- Wagner TC, Papalambros PY (1991) Optimal engine design using nonlinear programming and the engine system assessment model. Technical report, Ford Motor Company Scientific Research Laboratories, Dearborn, MI and Department of Mechanical Engineering at the University of Michigan, Ann Arbor, MI
- Warnes JJ, Ripley BD (1987) Problems with likelihood estimation of covariance function of spatial Gaussian processes. *Biometrika* 74(3):640–642