1st AIAA Multidisciplinary Design Optimization Specialist Conference, 18 - 21 April 2005, Austin, Texas

Update Strategies for Kriging Models for Use in Variable Fidelity Optimization

Shawn E. Gano* and John E. Renaud †

University of Notre Dame, Notre Dame, Indiana, 46556-5637, U.S.A.

Jay D. Martin‡

Applied Research Laboratory, State College, PA 16804, U.S.A.

Timothy W. Simpson§

The Pennsylvania State University, University Park, PA 16802, U.S.A.

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 each iteration. One solution to this problem would be to replace the kriging models with traditional Taylor series response surface models. Kriging models, however, have been shown to provide good approximations of computer simulations that incorporate larger amounts of data, resulting in better global accuracy. In this paper two metamodel update management schemes (MUMS) are proposed to reduce the cost of using kriging models sequentially by updating the kriging model parameters only when they produce a poor approximation. The two schemes differ in how they determine when the parameters should be updated. The first method uses ratios of likelihood values (L-MUMS), which are computed based on the model parameters and the data points used to construct the kriging model. The second 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 methods: an internal combustion engine sizing problem and a control-augmented structural design problem. The results indicate that the L-MUMS approach does not perform well. The TR-MUMS approach, however, was found to be 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 every iteration. It was also found that in trust region-based methods, 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 effecting the overall convergence rate, which, in turn, results in a faster execution time.

Nomenclature

β	Multiplicative scaling function
Δ	Trust region size
ϵ_f	Objective function convergence tolerance

*Ph.D. Candidate, Aerospace and Mechanical Engineering, sgano@nd.edu, Student Member AIAA.

†Professor, Aerospace and Mechanical Engineering, John.E.Renaud.2@nd.edu, Associate Fellow AIAA.

‡Research Assistant, Manufacturing Systems, P.O. Box 30, jdm111@psu.edu, Member AIAA.

§Associate Professor, Departments of Mechanical & Nuclear Engineering and Industrial & Manufacturing Engineering, 329 Leonhard Bldg., tws8@psu.edu, Member AIAA.

ϵ_x	Design variable convergence tolerance
γ	Additive scaling function
$\mathbf{R}$	Correlation matrix
$\mathbf{r}$	Correlation vector
ρ	Trust region ratio
σ^2	Variance
θ	Correlation or kriging model parameters
ϱ	Likelihood ratio
A_i	The ith kriging model management parmeter
K	Optimization scheme used to determine the kriging model parameters
L	Likelihood function
l	Log-likelihood function
MSE	Mean squared error
n_s	Number of sample sites used in kriging model
n_v	Number of variables used in kriging model
B	Global kriging trend function
P	l_1 penalty function
R	Correlation function
y	Function modeled using kriging
Z	Stochastic process model in kriging

Subscripts

n	Current iteration number
$scaled$	Scaled low fidelity value

Superscripts

$\hat{}$	Predicted value or function
^{T}	Transpose operator

I. 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 both construct and evaluate. However, global metamodels have been successfully used as approximations of the more expensive computer simulations in the design process.[1,2] 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.[3] The computational expense arises from having to re-compute 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[4] 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 Sacks *et al.*[3] and called Design and Analysis of Computer Experiments (DACE). Since this time it has been widely used to approximate computer models.[5–10] Kriging is flexible and quite robust in approximating complex multi-dimensional functions, making it well suited for this application.

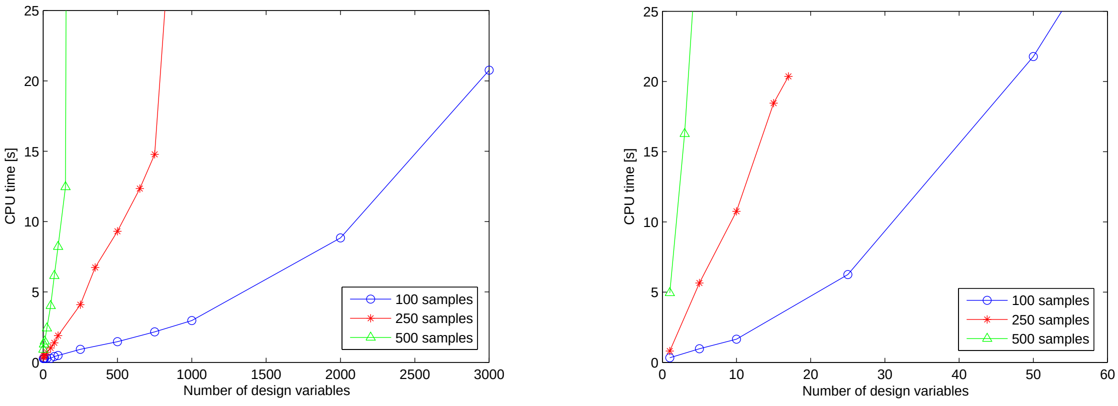
Kriging methods have been used to model the response of many engineering systems, including the design of a low-boom business jet by Chung and Alonso[11] and design of an aerospike nozzle by Simpson *et al.*[12] Martin and Simpson conducted a study on using kriging models to approximate deterministic computer models and discussed the applicability of various kriging variants.[13,14]

In the optimization field kriging models have mostly served as surrogate models. However, they have also been 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.*,[5] Jones *et al.*,[15] and Sasena *et al.*[16] all used kriging models as surrogates which were managed through the course of the optimization. Gano *et al.*[6] 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 approximate the high fidelity model. This scaling approach combined with a trust region managed scheme provably converges to the solution of the higher fidelity model with the intent of saving computational resources.

Kriging is typically used as an interpolation process; however, 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 as well as the number of sample points used to construct the kriging model. There are two main methods of estimating these parameters:[14] 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,[17] Warnes and Ripley,[18] and Martin and Simpson[13] 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[19] 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?

Because of the sequential nature of the variable fidelity optimization process, the kriging models are rebuilt to include new information each iteration. This process of rebuilding the kriging model by refitting the model parameters can be quite costly, as shown in Figure 1. The figure shows the exponential 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 Figure 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 two metamodel update management schemes (MUMS) are proposed to determine when the kriging model parameters should be updated; the methods are based on the ratio of successive likelihood values (L-MUMS) and on using the trust region ratio value (TR-MUMS). This research also determines if such management schemes 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 II) along with the local and global optimization approaches used to fit the kriging model parameters (Sections II.A - II.C). Then a brief overview of the variable fidelity method used to test the kriging model update strategies is given (Section III). Next, a description of the update schemes is provided (Section IV) as well as the numerical procedure for their evaluation (Section V). The update strategies are then demonstrated on an internal combustion engine design problem and a control-augmented structural design problem (Section VI). Finally, conclusions and future work are given (Section VII).



(a) Fixed model parameters

(b) Optimized model parameters

Figure 1. Relative cost of building kriging models with different numbers of samples and design variables.

II. Ordinary Kriging Using Maximum Likelihood Estimation

Ordinary kriging can be used to scale a low fidelity model to match a higher fidelity model in the variable fidelity optimization framework in Section III. The kriging model is built using all of the points in which the models have been evaluated. The method for constructing a kriging model is briefly explained in the rest of this section.

Kriging begins by estimating an unknown function, y, with the form[15, 20, 21]

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

The term B is an unknown constant or global trend function, typically linear or quadratic. $Z(x)$ is the model of a stochastic process with a mean of zero, a variance σ^2, and a co-variance of:

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

where $\mathbf{R}$ is the correlation matrix, R is the correlation function which is 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,[7, 12] the Gaussian function is by far the most popular and is also used in this work. It is defined as

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

where θ_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 kth 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 notation in this paper that distinguishes between the true value and an estimated value uses a hat to denote the estimated values, such as y is the true response value and $\hat{y}$ is the estimated value. The mean squared error is defined as the square of the expected value of the difference between the real response and the approximated one at any point. Mathematically this is stated as

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

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

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

where $\mathbf{y}$ is the vector of responses (objective values) to the sample locations $\{\mathbf{x}^1, \ldots, \mathbf{x}^{n_s}\}$ and $\mathbf{f}$ is a constant vector of all ones with length n_s. In Eq. (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 θ_k must both be found, as $\mathbf{r}$ and $\mathbf{R}$ depend on θ_k. The correlation vector is

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

The unknown parameters θ_k are found using maximum likelihood estimation.[7] This approach uses the likelihood of the assumed Gaussian computer model with kriging parameters of θ_k given observations $\mathbf{y}$. This likelihood is defined as,

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

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

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

By setting the derivatives of Eq. (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:

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

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

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

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

This is a n_v dimensional optimization problem which is well posed. Notice that a kriging model can be built for any values of θ_k; this optimization ensures the *best choice* for the θ_k values. In practice when solving for θ_k, $\mathbf{R}$ may become badly scaled; this is overcome using various numerical techniques as described by Lophaven *et al.*[22, 23]

A. Constructing the Kriging Model Using a Quasi-Newton Method

To solve the MLE problem in Eq. (11), a gradient-based or local optimization approach can be used. The main benefit of using such methods over stochastic ones is that they typically require many fewer function calls to converge. These methods make no claim or attempt at finding global solutions, only local optima, and have difficulty in large flat regions that may be occur in the likelihood function.

For this study a Quasi-Newton Broyden[24]-Fletcher[25]-Goldfarb[26]-Shanno[27] (BFGS) method is used in the form of MATLAB's *fminunc* function. Quasi-Newton methods successively minimize a second-order model of the objective function. The second-order information is approximated using successive gradient information using the BFGS update.

B. Constructing the Kriging Model Using Pattern Search

In the DACE toolbox for MATLAB developed by Lophaven *et al.*[22,23] a pattern search method is used to find the optimal kriging parameters. They specifically use the Hooke and Jeeves method, as described by Kowalik and Osborne,[28] using relative changes in the parameters instead of absolute changes. While this method does not use gradient information directly, it is still a local optimization method. Pattern search methods tend to be better suited for highly nonlinear or discontinuous functions, which may give this method an edge over a gradient-based approach, but tend to be less efficient when the function is first-order continuous. The pattern search will be compared to the gradient-based method (quasi-Newton) and the simulated annealing approach, described next, in the numerical experiments.

C. Constructing the Kriging Model Using Adaptive Simulated Annealing

In order to provide a more robust optimization of the likelihood function than the gradient-based method, given the multimodal and long near-optimal ridge features of the likelihood function, adaptive simulated annealing (ASA) is used. Adaptive simulated annealing was developed by Ingber[29–32] and is a more efficient and more robust implementation of simulated annealing. Simulated annealing[33] includes a temperature schedule for efficient searching by Kirkpatrick *et al.*[34] Other researchers have also independently developed a similar method.[35,36] Adaptive simulated annealing differs in the fact that typically fixed algorithmic parameters are allowed to adapt to each problem.

Simulated annealing, while more robust in finding the solution to the MLE problem, is also much more expensive in terms of function calls as it is a Monte Carlo method.[12] The expense of each function call in constructing the kriging model involves finding the inverse and/or determinate of the correlation matrix.

D. Comparison Between Optimizers When Building a Kriging Model: 1D Example

Using a simple one-dimensional function allows for comparing the kriging models produced with the different optimizers when solving the MLE problem. The sine function is used as the target function to be approximated. The kriging models are built using four sample points, $x = \{1.0,\ 1.05,\ 3.0,\ 3.05\}$, and $\mathbf{y}$ is the set or responses of $y(x) = \sin(x)$. The initial starting point was taken to be $\theta_0 = 5.5$. This is not a very good initial guess given the scale of the problem; however, this is just a demonstration of what could happen in a larger problem where little is known about choosing a good initial point. Figure 2 shows the likelihood function given the set of sample points. The figure shows the likelihood is very flat in the range $\theta = [1.5,\ 5.5]$ with a local maximum near $\theta = 3$ and a global maximum near $\theta = 0.4$.

The resulting kriging models constructed using no optimization and the three different optimizers are shown in Figure 3. The model that used no optimization produced the worst approximation, as expected. The two local optimization methods, Quasi-Newton and pattern search, produced very similar approximations. The best approximation of the sine function was produced by using the ASA optimizer.

In Table 1 the resulting value of θ from each optimization is given along with the final likelihood value and the number of likelihood evaluations needed for convergence (L Evals). The results in the table show that the two local optimizers converged to the local maxima, which is why their kriging models were similar. The ASA method converged to the global solution, giving a better approximation to the sine function. However, the ASA method used orders of magnitude more likelihood evaluations to converge; so, the better approximation was obtained at a much higher computational cost. The Quasi-Newton and pattern search had similar numbers of likelihood evaluations. The Quasi-Newton optimizer required a few more evaluations

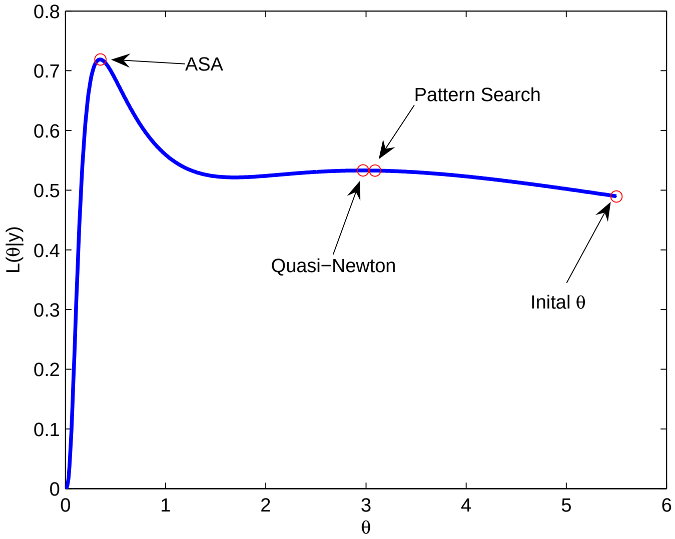


Figure 2. The likelihood function of the one dimensional example.

Table 1. Optimization Results for the One Dimensional Example

Optimizer	θ	$L(\theta\|\mathbf{y})$	L Evals
None	5.50	0.4894	1
Pattern Search	3.09	0.5328	9
Quasi-Newton	2.97	0.5329	20
ASA	0.35	0.7190	364

because it used finite differencing to determine needed gradient information.

In this example if the starting point was a little better, for example around $\theta = 1$, then all methods would have converged to the same point, the global optima. The relative number of function evaluations would have still been roughly the same; therefore, one would wonder why ASA should be used. This observation raises an interesting point: can it somehow be gauged whether or not an initial starting point for θ is close to the global optima? Then it would be possible to choose either a local optimizer or a global one to solve the MLE problem in order to construct the best approximation using the least amount of likelihood evaluations. This idea could be used in optimization algorithms where kriging models are built sequentially, as in variable fidelity optimization, where only a few points are added to the sample space each time the kriging model is reconstructed. It is also possible that no optimization is needed if the best θ values do not change much when adding a few more samples. More rigorous frameworks are proposed in Section IV drawing from these observations. The next section describes the variable fidelity framework that uses the successive kriging models.

III. Variable Fidelity Optimization

The typical framework for variable fidelity optimization is depicted in Figure 4 and is based, in part, on work done by Alexandrov[37] and is fully described by Gano *et al.*[6] 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 following process describes the basic steps of the framework:

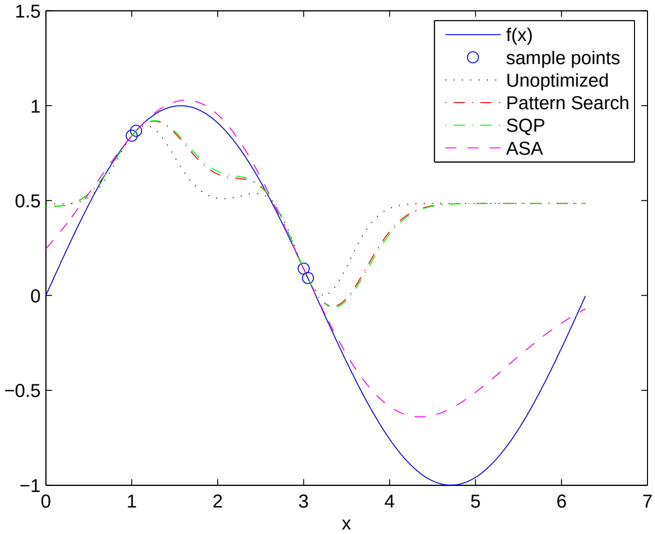


Figure 3. One-dimensional example comparing the kriging models produced by the different optimization methods.

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, $\mathbf{x}_n$.

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.

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,[38] three optimizers were compared: augmented Lagrangian method, multilevel algorithms for large-scale constrained optimization (MAESTRO)[39] (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 and scaled low fidelity models. The penalty function is defined as

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

where μ 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: In order to guarantee convergence of the variable fidelity optimization framework, a trust region model management strategy is employed.[40] This method provides a means for adaptively managing the allowable move limits for the approximate design space. Originally these methods

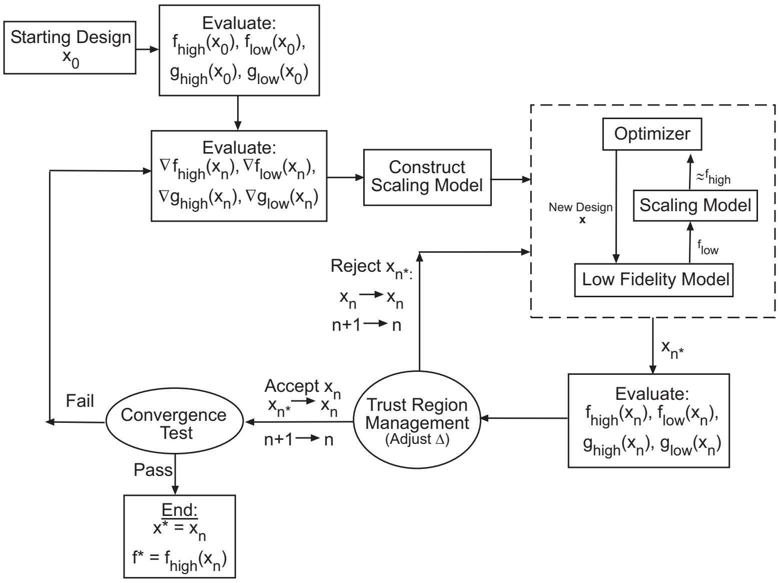


Figure 4. Varible fidelity framework flowchart.

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, ρ_n, is calculated at this new point:

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

where $P()_{high}$ and $P()_{scaled}$ are the l_1 penalty functions for the high and scaled low fidelity models and the point $\mathbf{x}_n$ was the initial point of the optimization. Notice that by definition $P(\mathbf{x}_n)_{scaled} = P(\mathbf{x}_n)_{high}$, because the scaled low fidelity model matches the high fidelity model at that point. This 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 the reasoning behind using the l_1 penalty function.
The trust region size is governed by the following standard rules:[41,42]

$$\Delta_{n+1} = \left\{ \begin{array}{ccl} c_1\Delta_n & : & \rho_n \leq R_1 \vee \rho_n > R_3 \\ \Delta_n & : & R_1 < \rho_n < R_2 \\ \Gamma\Delta_n & : & R_2 \leq \rho_n \leq R_3 \end{array} \right. . \tag{14}$$

where $\Gamma = c_2$ if $\|x_k^* - x_{c_k}\|_\infty = \Delta_k$ 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, ρ represents how good of an approximation our scaled low fidelity model is compared to the high fidelity model. If ρ is near 1, the approximation is quite good. If ρ is near zero, then the approximation is not as good, but it still captures the minimization trend. If ρ 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: The convergence of the entire framework is governed by satisfying the Karush-Kuhn-Tucker conditions as is shown by Rodríguez *et al.*[43] and Conn *et al.*[44] For the implementation used in this research the convergence was determined by the following two inequalities:

$$
\begin{aligned}
f_{high}(\mathbf{x}_n) - f_{high}(\mathbf{x}_{n-1}) &< \epsilon_f, \text{ and} \qquad (15)\\
\|\mathbf{x}_n - \mathbf{x}_{n-1}\| &< \epsilon_x, \qquad (16)
\end{aligned}
$$

where ϵ_f and ϵ_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.

A. 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[45] based on Chang's[46] scaling function. The additive method was presented by Lewis and Nash.[47] Both methods are based on constructing an unknown function to update the lower fidelity model, which in turn, will approximate the higher fidelity model.

1. *Multiplicative Scaling*

A given set of high and low fidelity models, $f_{high}(\mathbf{x})$ and $f_{low}(\mathbf{x})$, can be matched by multiplying the low fidelity model by an unknown function $\beta(\mathbf{x})$, which is posed mathematically as

$$f_{high}(\mathbf{x}) = \beta(\mathbf{x}) f_{low}(\mathbf{x}). \qquad (17)$$

This scaling model was first proposed and used for approximating structural response by Chang *et al.*[46] Solving for the unknown multiplicative scaling function results in

$$\beta(\mathbf{x}) = \frac{f_{high}(\mathbf{x})}{f_{low}(\mathbf{x})}. \qquad (18)$$

From inspection of Eq. 18, it is clear that the function $\beta(\mathbf{x})$ is the scaling ratio of the high fidelity model to the low fidelity model, and when it is multiplied by the low fidelity model, the value of the high fidelity model is achieved.

2. *Additive Scaling*

A given set of high and low fidelity models, $f_{high}(\mathbf{x})$ and $f_{low}(\mathbf{x})$, can also be matched by adding the low fidelity model to an unknown function $\gamma(\mathbf{x})$. This is expressed mathematically as

$$f_{high}(\mathbf{x}) = f_{low}(\mathbf{x}) + \gamma(\mathbf{x}). \qquad (19)$$

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

$$\gamma(\mathbf{x}) = f_{high}(\mathbf{x}) - f_{low}(\mathbf{x}). \qquad (20)$$

From Eq. 20, 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. A similar function for the constraints can be developed in the same manner as Eqs. (19) and (20).

B. 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. In this investigation a new kriging-based scaling function is developed to improve the scaling between the different fidelity models on a more global scale. 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 first-order matching is obtained. With the inclusion of gradient information, first-order matching is achieved. First-order matching combined with the trust region model management strategy provides for a provably convergent framework.[37]

Building and rebuilding the kriging models takes extra time and memory storage; however, this added computational time and resource expenditure may be negligible compared to the evaluation of the high fidelity model. The method used to build the kriging models is described in the following section.

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.

IV. Metamodel Update Management Strategies For Sequentially Building Kriging Models

This section discusses the update strategies used to reduce the cost of having to optimize the kriging model parameters at each iteration of sequential approximation optimization (SAO) method-like variable fidelity optimization. Both methods use some means to try to measure how good a kriging model is and if it should be updated or not after new samples are included. The first method, L-MUMS, uses the likelihood value, which is measured directly from the kriging model itself. The second method, TR-MUMS, uses the trust region ratio value, which gives a measure of how well the approximation matches the true model.

A. L-MUMS

The likelihood value obtained from Eq.(7) when constructing a kriging model gives a measure of how likely it is that the values used for the model parameters, θ, are the best values. To compare different kriging models, a ratio of their likelihood values can be used. If, for instance, the optimal model parameters are found for a set of data samples and then later a few more data samples are included, the new likelihood could be compared to the previous value to determine if the same model parameters can be used. This assumes that the inclusion of new data points does not significantly change the likelihood function.

To save computational cost of optimizing the kriging model parameters for each iteration of the variable fidelity framework a likelihood ratio of the following form is proposed:

$$\varrho_n = \frac{L_n(\theta_n|\mathbf{y}_n)}{L_n(\theta_n|\mathbf{y}_{n-1})}, \tag{21}$$

where $\mathbf{y}_{n-1}$ is the data set of the previous kriging model update and $\mathbf{y}_n$ is the current data set. If the likelihood ratio is near unity then the model parameters do not need to be updated, but if the ratio indicates a small change in likelihood a local optimization method starting from the current model parameters could easily move to the new optima relatively inexpensively. On the other hand, if the likelihood ratio shows a large change in likelihoods after the new points have been included, then a stochastic optimizer (e.g., simulated annealing) might be required to account for the multi-modal behavior of the likelihood function. This updating management scheme can be mathematically stated as:

$$K_n = \begin{cases} \mathit{global} & : \quad \varrho_n \leq \frac{1}{A_1} \vee A_1 < \varrho_n \\ \mathit{local} & : \quad \frac{1}{A_1} < \varrho_n < \frac{1}{A_2} \vee A_2 < \varrho_n < A_1 \\ \mathit{none} & : \quad \frac{1}{A_2} < \varrho_n < A_2 \end{cases} , \tag{22}$$

where K_n is the optimization scheme used to determine the kriging model parameters for the nth iteration. This scheme has only two parameters which need to be tuned, A_1 and A_2. For the first iteration the kriging model parameters are found using any optimization approach.

B. TR-MUMS

An alternative approach to managing the updating of the kriging model parameters is to use the trust region ratio, ρ_n defined in Eq. (13), that 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. It, therefore, 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 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 is simple. The update scheme, in its most general form is as follows:

$$K_{n+1} = \begin{cases} \mathit{global} & : \quad \rho_n \leq A_1 \\ \mathit{local} & : \quad A_1 < \rho_n < A_2 \\ \mathit{none} & : \quad A_2 \leq \rho_n \leq A_3 \end{cases} . \tag{23}$$

The parameters A_1, A_2, and A_3 do not necessarily have to correspond to the values used in updating the trust region size in Eq. (14). This update scheme uses a global optimizer when the trust region ratio is far from one, a local optimizer when the ratio is moderately close to one, and no update when the ratio is near one. A simpler version of this scheme is one that only updates using a single optimizer. This reduces to a simpler scheme as follows:

$$K_{n+1} = \begin{cases} \mathit{update} & : \quad \rho_n \leq A_1 \\ \mathit{none} & : \quad \rho_n > A_1 \end{cases} . \tag{24}$$

V. 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 unduely 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 structure 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 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 to 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[48] were used as the low fidelity objectives and constraints for both problems. The were generated using a high fidelity Latin hypercube sampling[49,50] using the number of design variables squared data points. The computational costs of evaluating of 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.

A. 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,[51] and a robust variation of the problem was solved by McAllister and Simpson.[52] The engine analysis parameters were determined by Ford's Engine Assessment Model (ESA). A schematic for the flat head combustion chamber is shown in Figure 5. The design variables for this problem are the cylinder bore b, the compression ratio c_r, exhaust value diameter d_E, intake value diameter d_I, and the revolutions per minute at peak power w. The design problem is mathematically posed below.

$$\begin{aligned}
\text{minimize} \quad & f = K_0\left(\tfrac{\rho Q}{A_f}\eta_t\eta_v - FMEP\right)w && (25)\\
\text{subject to} \quad & K_1 N_c b - L_1 \leq 0, && \text{[min bore wall thickness]} \quad (26)\\
& \sqrt{\tfrac{4K_2V}{\pi N_c L_2}} - b \leq 0, && \text{[max engine height]} \quad (27)\\
& d_I + d_E - K_3 b \leq 0, && \text{[valve structure]} \quad (28)\\
& K_4 d_I - d_E \leq 0, && \text{[min valve diameter ratio]} \quad (29)\\
& d_E - K_5 d_I \leq 0, && \text{[max valve diameter ratio]} \quad (30)\\
& 9.428\times10^{-5}\tfrac{4V}{\pi N_c}\tfrac{w}{d_I^2} - K_6 C_s \leq 0, && \text{[max Mach index]} \quad (31)\\
& c_r - 13.2 + 0.045b \leq 0, && \text{[knock-limited compression ratio]} \quad (32)\\
& w - K_7 \leq 0, && \text{[max torque converter rpm]} \quad (33)\\
& 3.6\times10^6 - K_8 Q\eta_{tw} \leq 0. && \text{[fuel economy]} \quad (34)
\end{aligned}$$

The thermal η_t, volumetric η_v, and thermal at part load point efficiencies η_{tw}, are all functions of the design variables and are given in the original paper.[51] The original paper also includes expressions for the friction mean effective pressure (FMEP), density of the inlet charge ρ, 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 $K_i, i = \{0...12\}$ and $L_i, i = \{1,2\}$. The starting and optimal designs are given in Table 2 along with their respective objective function values.

Table 2. 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)	7500	6070
f (kW/L)	30.28	55.67

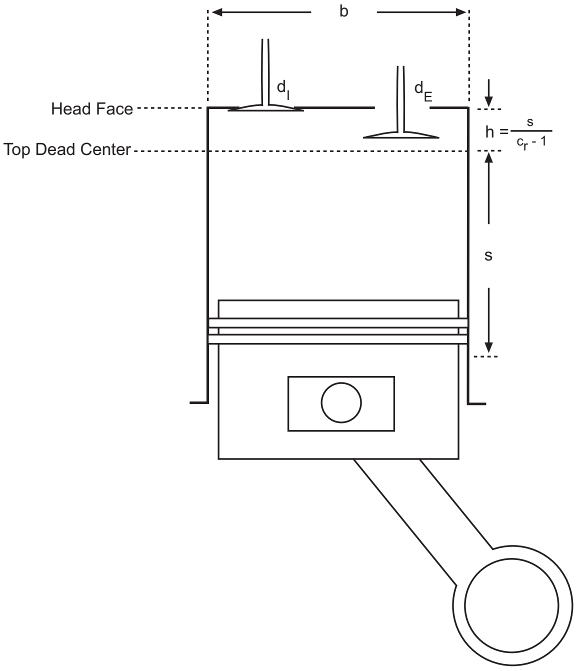


Figure 5. Combustion chamber geometry.

B. Control-Augmented Structure Design

The control-augmented structure design problem shown in Figure 6 was introduced by Sobieszczanski-Sobieski *et al.*[53] 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.

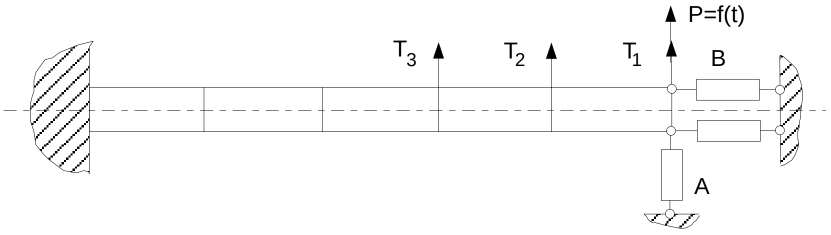


Figure 6. Cantilever beam with actuators.

The system analysis is comprised of two coupled contributing analyses as shown in Figure 7. The structures subsystem, $\mathrm{CA_s}$, consists of a finite element model of the beam where the natural frequencies and modes of the cantilever beam are computed. $\mathrm{CA_s}$ 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, $\mathrm{CA_c}$. 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 CAs 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 minimization is subjected to seven con-

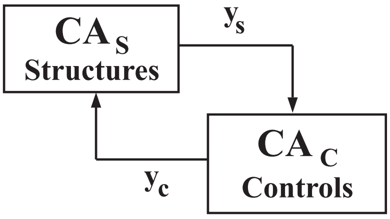


Figure 7. Dependency diagram of the Control-Augmented Structure design problem.

straints 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:

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

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, ω_1 is the first natural frequency, ω_2 is the second natural frequency, and σ is the static stress. The subscript a stands for the allowed value. The optimum for this problem is depicted in Table 3. The minimum weight, $W = 1493.6$ lbs occurs where 6 design variables are at their bounds.

Table 3. Starting and optimum designs for the Controls-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

VI. Results

The results found from optimizing the two demonstration problems using the numerical procedure outlined in the previous section are given in this section. An immediate observation was found when attempting to solve the two problems using L-MUMS. It was observed that for problems with nonlinear design spaces larger than a couple of design variables, the magnitude of the likelihood fluctuated by many orders of magnitude when just a few additional samples were added; this violated the assumption from which the update method was derived. This method caused the kriging model to be updated every iteration. Only the TR-MUMS results are given in the numerical studies for this reason. It was, therefore, concluded that the proposed likelihood ratio was not a good indicator of when to update the kriging model parameters.

A. 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 4. 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.

The first section of Table 4 shows the results for the variable fidelity optimization when the kriging model parameters were updated 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 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.

Table 4. Internal Combustion Engine Optimization Results

Optimizer	Iter	High Fn Calls	Low Fn Calls	L Evals	θ Updates
θ 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 (θ 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
θ 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	-	-	-

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 4. 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.*;[54] they noted the kriging best linear unbiased predictor (BLUP) 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 4. 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; and 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.

B. 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 5. A high fidelity function call required 1.5s to evaluate, the low fidelity model took 0.01s to evaluate, and to optimize the kriging models, on average, took 3s and 900s 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 is 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 5 shows the results when the kriging model parameters were updated each iteration. 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 5 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 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 for updating 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 every iteration. The results are given in the second section of Table 5. The reduction in likelihood evaluations, however, came at a cost of slightly increasing the number of high fidelity 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.

Table 5. **Control-Augmented Structure Optimization Results**

Optimizer	Iter	High Fn Calls	Low Fn Calls	L Evals	θ Updates	Wall Time (s)
θ 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 (θ 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
θ updated once						
ASA*	17	117	1,430	100,016	1	1,090
Pattern Search*	35	333	4,214	140	1	545
Standard single fidelity optimization						
SQP	26	340	-	-	-	510

* Did not converge to the optimum; the trust region size became too small.

VII. Summary, Conclusions, and Future Work

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. It was pointed out that the likelihood function, which is the objective in determining the kriging model parameters, can be multi-modal 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, have been used by researchers to address this problem. However, such methods required many orders of magnitude more likelihood evaluations to converge. Two metamodel update management schemes were proposed in this paper to reduce the cost of using kriging models that are sequentially updated. The first scheme, L-MUMS, used likelihood ratios to determine when to update the kriging model parameters. The second, 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 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 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. It was also found that the likelihood ratio was not a good measure of when the kriging model parameters need to be updated. However, the trust region ratio was found to be a good indicator, and its use resulted in fewer kriging model parameter updates and a lower total cost of optimization.

Future work on metamodel update management schemes could include using an improved trend function in the kriging model as recommended by Martin and Simpson.[14] Using a non-constant trend function will address the issues of multi-modality 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 multi-modality of the likelihood function. It also results faster optimization of the MLE problem when using gradient methods. Furthermore, work should be done on studying why the L-MUMS was so ineffective. One possible improvement could be to use a method more like the likelihood ratio test, using the logarithm of the likelihood and the Chi-square test.

Acknowledgments

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

[1]Booker, A. J., "Design and Analysis of Computer Experiments," *Proceedings of the 7th AIAA/USAF/NASA ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Vol. 1, St. Louis, MO, 1998, pp. 118–128.

[2]Simpson, T. W., Maurey, T. M., Korte, J. J., and Mistree, F., "Kriging Metamodels for Global Approximation in Simulation-Based Design," *Engineering sith Computers*, Vol. 17, No. 2, 2001, pp. 129–150.

[3]Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P., "Design and Analysis of Computer Experiments," *Statistical Science*, Vol. 4, 1989, pp. 409–423.

[4]Matheron, G., *The Theory of Regionalized Variables and its Applications.*, Ecole des Mines, Fontainebleau, 1971.

[5]Booker, A. J., Dennis, J. E., Frank, P. D., Serafini, D. B., Torczon, V., and Trosset, M. W., "A Rigorous Framework for Optimization of Expensive Function by Surrogates," *Structural Optimization*, Vol. 17, No. 1, 1999, pp. 1–13.

[6]Gano, S. E., Renuad, J. E., and Sanders, B., "Variable Fidelity Optimization Using a Kriging Based Scaling Function," *Proceedings of the 11th AIAA/ISSMO Multidisciplinary Analysis & Optimization Conference*, No. AIAA 2004-4460, Albany, NY, 30 August - 1 September 2004.

[7]Giunta, A. A. and Watson, L. T., "A Comparison of Approximation Modeling Techniques: Polynomial Versus Interpolating Models," No. AIAA-98-4758, 1998.

[8]Chung, H.-S. and Alonso, J. J., "Design of a Low-Boom Supersonic Business Jet Using Cokriging Approximation Models," *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, No. AIAA 2002-5598, Atlanta, GA, 4-6 September 2002.

[9]Simpson, T. W., Peplinski, J. D., Koch, P. N., and Allen, J. K., "Metamodels for Computer-based Engineering Design: Survey and recommendations," *Engineering with Computers*, Vol. 17, 2001, pp. 129–150.

[10]Simpson, T. W., Booker, A. J., Gosh, D., and Gunita, A. A., "Approximation Methods in Multidisciplinary Analysis and Optimization: A Panel Discussion," *Structural and Multidisciplinary Optimization*, Vol. 27, No. 5, 2004, pp. 302–313.

[11]Chung, H.-S. and Alonso, J. J., "Using Gradients to Construct Cokriging Approximation Models for High-Dimensional Design Optimization Problems," *40th AIAA Aerospace Sciences Meeting and Exhibit*, No. AIAA 2002-0317, Reno, NV, 14-17 January 2002.

[12]Simpson, T. W., Mauery, T. M., Korte, J. J., and Mistree, F., "Kriging Metamodels for Global Approximation in Simulation-Based Multidisciplinary Design Optimization," *AIAA Journal*, Vol. 39, No. 12, 2001, pp. 2233–2241.

[13]Martin, J. D. and Simpson, T. W., "A Study on the Use of Kriging Models to Approximate Deterministic Computer Models," *Proceedings of ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, No. DET2003/DAC-48762, Chicago, IL, 2-6 September 2003.

[14]Martin, J. D. and Simpson, T. W., "On the Use of Kriging Models to Approximate Deterministic Computer Models," *AIAA Journal*, 2005, In Press.

[15]Jones, D. R., Schonlau, M., and Welch, W. J., "Efficient Global Optimization of Expensive Black-Box Functions," *Journal of Global Optimization*, Vol. 13, 1998, pp. 455–492.

[16]Sasena, M. J., Papalmbros, P., and Goovaerts, P., "Exploration of Metamodeling Sampling Criteria for Constrained Global Optimization," *Engineering Optimization*, Vol. 34, No. 3, 2002, pp. 263–278.

[17]Ripley, B. D., *Statistical Inference For Spatial Processes*, Cambridge University Press, 1988.

[18]Warnes, J. J. and Ripley, B. D., "Problems with Likelihood Estimation of Covariance Function of Spatial Gaussian Processes," *Biometrika*, Vol. 74, No. 3, 1987, pp. 640–642.

[19]Martin, J. D. and Simpson, T. W., "A Monte Carlo Simulation of the Kriging Model," *Proceedings of the 11th AIAA/ISSMO Multidisciplinary Analysis & Optimization Conference*, No. AIAA 2004-4483, Albany, NY, 30 August - 1 September 2004.

[20]Denison, D. G. T., Holmes, C. C., Mallick, B. K., and Smith, A. F. M., *Bayesian Methods for Nonlinear Classification and Regression*, John Wiley & Sons, Inc., 2002.

[21]Stein, M. L., *Interpolation of Spatial Data: Some Theory for Kriging*, Springer-Verlag, 1999.

[22]Lophaven, S. N., Nielsen, H. B., and Søndergaard, J., "DACE a Matlab Kriging Toolbox," Tech. Rep. IMM-REP-2002-12, Technical University of Denmark, DK-2800 Kongens Lyngby - Denmark, 1 August 2002, version 2.0.

[23]Lophaven, S. N., Nielsen, H. B., and Søndergaard, J., "Aspects of the Matlab Toolbox DACE," Tech. Rep. IMM-REP-2002-13, Technical University of Denmark, DK-2800 Kongens Lyngby - Denmark, 2002.

[24]Broyden, C. G., "The Convergence of a Class of Double-rank Minimization Algorithms," *J. Inst. Maths. Applics.*, Vol. 6, 1970, pp. 76–90.

[25]Fletcher, R., "A New Approach to Variable Metric Algorithms," *Computer Journal*, Vol. 13, 1970, pp. 317–322.

[26]Goldfarb, D., "A Family of Variable Metric Updates Derived by Variational Means," *Mathematics of Computing*, Vol. 24, 1970, pp. 23–26.

[27] Shanno, D. F., "Conditioning of Quasi-Newton Methods for Function Minimization," *Mathematics of Computing*, Vol. 24, 1970, pp. 647–656.

[28] Kowalik, J. and Osborne, M. R., *Methods for Unconstrained Optimization Problems*, American Elsevier, 1968.

[29] Ingber, L., "Very Fast Simulated Re-Annealing," *Journal of Mathematical and Computer Modelling*, Vol. 12, 1989, pp. 967–973.

[30] Ingber, L., "Genetic Algorithms and Very Fast Simulated Reannealing: A Comparison," *Journal of Mathematical and Computer Modelling*, Vol. 16, No. 11, 1992, pp. 87–100.

[31] Ingber, L., "Simulated Annealing: Practice Versus Theory," *Journal of Mathematical and Computer Modelling*, Vol. 18, No. 11, 1993, pp. 29–57.

[32] Ingber, L., "Adaptive Simulated Annealing (ASA): Lessons Learned," *Journal of Control and Cybernetics*, Vol. 25, No. 1, 1996, pp. 33–54.

[33] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E., "Equation of State Calculations by Fast Computing Machines," *Journal of Chemical Physics*, Vol. 21, No. 6, 1953, pp. 1087–1092.

[34] Kirkpatrick, S., Gelatt, C. D. J., and Vecchi, M. P., "Optimization by Simulated Annealing," *Science*, Vol. 220, No. 4598, 1983, pp. 671–680.

[35] Cerny, V., "A Thermodynamical Approach to the Travelling Salesman Problem: An Efficient Simulation Algorithm," Tech. rep., Comenius University, Bratislava, Czechoslovakia, 1982, Report.

[36] Pincus, M., "A Monte Carlo Method for the Approximate Solution of Certain Types of Constrained Optimization Problems," *Operations Research*, Vol. 18, 1970, pp. 1225–1228.

[37] Alexandrov, N. M., Dennis, J. E., Lewis, R. M., and Torczon, V., "A Trust Region Framework for Managing the Use of Approximation Models in Optimization," No. ICASE Report 97-50, NASA/CR-201745, 2001.

[38] Alexandrov, N. M., Lewis, R. M., Gumbert, C. R., Green, L. L., and Newman, P. A., "Optimization with variable-fidelity models applied to wing design," *Proceedings of the 38th AIAA Aerospace Sciences Meeting and Exhibit*, No. AIAA 2000-0841, Reno, Nevada, 10-13 January 2000.

[39] Alexandrov, N. M., Nielsen, E. J., Lewis, R. M., and Anderson, W. K., "First-Order Model Management with Variable-Fidelity Physics Applied to Multi-Element Airfoil Optimization," No. AIAA paper 2000-4886, September 2000.

[40] Conn, A. R., Gould, N. I. M., and Toint, P. L., "Global Convergence of a Class of Trust Region Algorithms for Optimization with Simple Bounds," *SIAM Journal of Numerical Analysis*, Vol. 25, No. 2, 1988, pp. 433–464.

[41] Giunta, A. A. and Eldred, M. S., "Implementation of a Trust Region Model Management Strategy in the DAKOTA Optimization Toolkit," No. AIAA-2000-4935, 2000.

[42] Rodriguez, J. F., Pérez, V. M., Padmanabhan, D., and Renaud, J. E., "Sequential Approximate Optimization Using Variable Fidelity Response Surface Approximations," *Structural and Multidisciplinary Optimization*, Vol. 22, 2001, pp. 24–34, Published by Springer-Verlag.

[43] Rodríguez, J. F. and Renaud, J. E., "Convergence of Trust Region Augmented Lagrangian Methods Using Variable Fidelity Approximation Data," *Structural Optimization*, Vol. 15, 1998, pp. 141–156.

[44] Conn, A. R., Gould, N. I. M., and Toint, P. L., *Trust-Region Methods*, Society for Industrial and Applied Mathematics and Mathematical Programming Society, 2000.

[45] Alexandrov, N. M. and Lewis, R. M., *Large-Scale PDE-Constrained Optimization*, Vol. 30, 2003, Series: Lecture Notes in Computational Science and Engineering.

[46] Chang, K. J., Haftka, R. T., Giles, G. L., and Kao, P.-J., "Sensitivity-based Scaling for Approximating Structural Response," *Journal of Aircraft*, Vol. 30(2), March-April 1993, pp. 283–288.

[47] Lewis, R. M. and Nash, S. G., "A Mulitgrid Approach to the Optimization of Systems Governed by Differential Equations," No. AIAA-2000-4890, 2000.

[48] Myers, R. H. and Montgomery, D. C., *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, Wiley, 1995.

[49] Iman, R. L., Helton, J. C., and Cambpell, J. E., "An Approach to Sensitivity Analysis of Computer Models, Part I: Introduction, input variable selection and preliminary variable assessment," *Journal of Quality Technology*, Vol. 13, 1981, pp. 174–183.

[50] Iman, R. L., Helton, J. C., and Cambpell, J. E., "An Approach to Sensitivity Analysis of Computer Models, Part II: Ranking of input variables, response surface validation, distribution effect and technique synopsis," *Journal of Quality Technology*, Vol. 13, 1981, pp. 232–240.

[51] Wagner, T. C. and Papalambros, P. Y., "Optimal Engine Design Using Nonlinear Programming and the Engine System Assessment Model," Tech. rep., Ford Motor Company Scientific Research Laboratories, Dearborn, MI. Department of Mechanical Engineering at the University of Michigan, Ann Arbor, MI., 1991.

[52] McAllister, C. D. and Simpson, T. W., "Multidisciplinary Robust Design Optimization of an Internal Combustion Engine," *Transactions of the ASME - Journal of Mechanical Design*, Vol. 125, 2003, pp. 124–130.

[53] Sobieszczanski-Sobieski, J., Bloebaum, C. L., and Hajela, P., "Sensitivity of Control-Augmented Structure Obtained by a System Decomposition Method," *AIAA Journal*, Vol. 29, No. 2, 1991, pp. 264–270.

[54] Lim, Y. B., Sacks, J., J., S. W., and Welch, W. J., "Design and Analysis of Computer Experiments When the Output is Highly Correlated Over the Input Space," Tech. Rep. 62, National Institute of Statistical Sciences, 19 T.W. Alexander Dr., PO Box 14006, Research Triangle Park, NC 27709-4006, August 1997.