AIAA JOURNAL
Vol. 44, No. 10, October 2006

Design and Analysis of "Noisy" Computer Experiments

Alexander I. J. Forrester,* Andy J. Keane,† and Neil W. Bressloff‡
University of Southampton, Southampton, Hampshire, SO17 1BJ England, United Kingdom

DOI: 10.2514/1.20068

Recently there has been a growing interest in using response surface techniques to expedite the global optimization of functions calculated by long running computer codes. The literature in this area commonly assumes that the objective function is a smooth, deterministic function of the inputs. Yet it is well known that many computer simulations, especially those of computational fluid and structural dynamics codes, often display what one might call numerical noise: rather than lying on a smooth curve, results appear to contain a random scatter about a smooth trend. This paper extends previous optimization methods based on the interpolating method of *kriging* to the case of such noisy computer experiments. Firstly, we review how the kriging interpolation can be modified to filter out numerical noise. We then show how to adjust the estimate of the error in a kriging prediction so that previous approaches to optimization, such as the method of maximizing the expected improvement, continue to work effectively. We introduce the problems associated with noise and demonstrate our approach using computational fluid dynamics based problems.

I. Introduction

THE use of response surface models (RSMs, also known as surrogate, meta, or approximation models) in optimization is becoming increasingly popular. The RSM is not in itself an optimizer, but rather a tool for increasing the speed of optimization. Instead of making direct calls to an expensive computer code, an optimization routine takes values from a cheap surrogate model of the computer code. The popularity of such methods has probably increased due to the development of approximation methods that are able to capture the shape of multimodal landscapes. Low-order polynomials [1] are notorious for predicting erroneous optima in complex functions, whereas the more advanced method of kriging[§] [4–6] has been shown by Jones et al. [7] and others to be a robust, though complex, method able to predict even the most deceptive of global optima.

Although kriging approximations often give good predictions, an optimization based on predictions from any RSM can only be guaranteed to find the optimum of the RSM, which may not in fact agree with the optimum of the computer simulations. The use of kriging is attractive because, not only can it give good predictions of complex landscapes, it also provides a credible estimate of the possible error in these predictions. These error estimates make it possible to make tradeoffs between sampling where the current prediction is good (local exploitation) and sampling where there is high uncertainty in the function value (global exploration). There is a growing literature on how to use *both* the kriging prediction and estimated error to choose points at which to run new computer experiments in order to refine the RSM when performing optimization [8–10]. To date, however, all of this literature has assumed that the computer simulation is best approximated by an interpolating surface, that is, a surface that goes through all the data points. As we explain shortly, certain computer simulations, such as those in crashworthiness and computational fluid dynamics (CFD), exhibit a kind of numerical noise, which makes it more natural to fit a regressing kriging surface that extracts a smooth trend from the data and filters out this noise (i.e., does not go through the data points). The focus of this paper is how to extend the optimization methods previously used for interpolating kriging to this regressing kriging.

Computer experiments have outputs that are deterministic functions of their inputs, and as such require different techniques from physical experiments. The initial difference in approaching these two scenarios is in the selection of which experiments are to be performed to provide the data from which to build our RSM [the design of experiment (DOE)], that is, which experiments will give us the best possible idea of how an output (our objective or cost function) depends on a number of inputs (our optimization variables). Two identical physical experiments are likely to produce different results because, inevitably, there are factors affecting the output that vary beyond our control. As such, repeated experiments are normally included in any DOE used for physical experiments in order to average out this experimental error. Deterministic computer experiments have no factors varying beyond our control. No repetition is required, because the same inputs will always yield the same result and so as many different experiments as possible are included in the DOE.

The analysis of the experiments, in this case the construction of a surrogate model, is also different. The random error in physical experiments lends itself to the use of a regression model as a noise filter, whereas computer experiments are often approximated using interpolating models, such as kriging or other radial basis function methods. This is an appropriate course of action in many cases. If, however, the computer experiment is based on the simulation of physical phenomena, using iterative and/or discretised schemes, the output may have traits in common with a true physical experiment. Although an identical experiment will yield the same result, inputs immediately adjacent to this point may produce a quite different result, due to perturbations in the numerical solution of the physical phenomena (e.g., changes in a computational mesh, particularly when performing a fast simulation through the use of a coarse mesh). Such random deviations from the expected smooth response are what we categorize as noise in this paper. An example of this numerical noise is seen in Fig. 1, where, as the shape of an airfoil is changed, the CFD predicted drag shows fluctuations about a smooth trend (the airfoil problem setup and the origin of this noise are discussed in Sec. II). Although the analysis is still deterministic and so the DOE should have no repeated experiments, a regressing (smoothing) surrogate is now appropriate in order to filter out this different type of experimental error.

Received 14 September 2005; revision received 20 April 2006; accepted for publication 20 April 2006.

*Research Fellow, Computational Engineering and Design Group.
†Professor of Computational Engineering, Computational Engineering and Design Group.
‡Senior Research Fellow, Computational Engineering and Design Group.

§So called by Matheron [2] after D. G. Krige, a South African mining engineer who developed the method in the 1950s for determining ore grade distributions based on core samples [3].

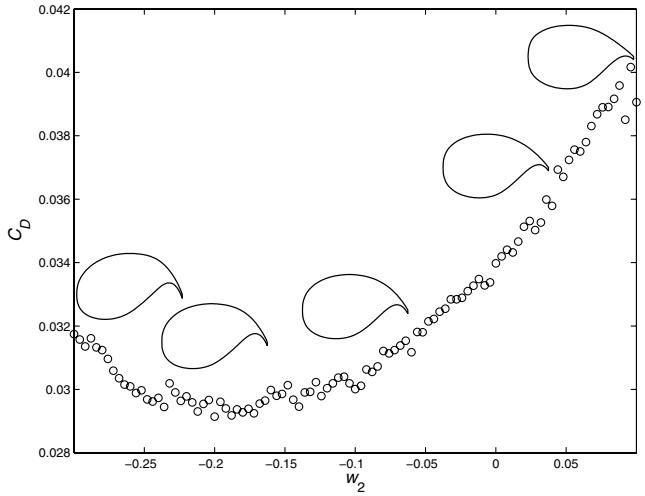


Fig. 1 Noise due to changing mesh. Airfoil geometries displayed (with exaggerated thickness) for $w_2 = -0.3, -0.2, -0.1, 0$, and 0.1.

The identification and filtering of noise through the use of RSM techniques is well documented [11–14]. These references generally deal with the removal of noise through the use of polynomial regression RSMs. By using an RSM to filter noise we are assuming that the function is smooth, but polynomial approximations go further to assume that the function takes on a specific form, for example, quadratic. This approach can naturally lead to underfitting, with key trends in the data being filtered out along with the noise. Using a regressing kriging model allows us to filter noise without having to guess the functional form of the underlying smooth trend. Although kriging regression gives a good noise filtering model, the error estimates are no longer appropriate for use when choosing points at which to run new computer experiments. A popular way of combining local exploitation and global exploration into a single figure of merit is the expected improvement criterion, which we examine in Sec. III. Locatelli [15] has proven that a search based on running new experiments at points of maximum expected improvement of a kriging interpolation converges toward the global optimum. However, if noise is filtered using a regressing kriging model, the error estimates no longer exhibit the key property required by Locatelli's proof, namely, that the error is zero at a sampled point. We present a method for calculating error estimates that restores this property and assures convergence toward the global optimum.

We begin in the next section by examining what causes noise in the output of computer experiments and, in particular, CFD simulations. We then briefly examine the kriging equations (with derivations given in the Appendix) for both prediction and optimization. Section IV examines the process of, and problems associated with, optimizing noisy functions, first using interpolation, then regression, and finally using our method of reinterpolation, which improves convergence when using regression. We then apply these techniques to an example airfoil optimization, before drawing conclusions in the final section.

II. On the Origin of "Noise" in Computer Experiments

The term noise usually refers to random fluctuations in the output of an experiment, which are unrelated to the deliberately varied inputs. We expect to see noise in physical experiments because there are many factors outside of our control. Here we are concerned with the use of deterministic computer experiments where this explanation of noise does not hold true, because a given input will always produce the same output. There is likely to be error in the output of a simulation of physical phenomena, but this error is repeatable. However, variation in the output due to fluctuations in the error from experiment to experiment as the inputs are varied slightly *appears* to be noise. For example, consider the use of CFD to calculate aerodynamic forces; error occurs in the result due to three main reasons: 1) discretization error, 2) incomplete convergence, and 3) inaccurate application of boundary conditions (e.g., iterating during the solution toward a fixed value of lift). Roundoff error due to finite machine accuracy will also produce errors, but to a lesser extent. We now demonstrate the noise these errors produce via a one-variable airfoil optimization.

Consider an airfoil where the drag coefficient C_D is to be minimized subject to a fixed lift constraint $C_L = 0.6$ at freestream Mach number $M_\infty = 0.73$ and standard atmosphere conditions at 10,000 m. Inviscid flow simulations are performed using the commercial CFD package Fluent [16] with a rather coarse 13,000 cell unstructured mesh and first-order accurate solution scheme.

The airfoil is defined by five orthogonal shape functions [17] and a thickness-to-chord ratio $t{:}c$. The first function, which represents the shape of a NASA supercritical SC(2) airfoil [18], and $t{:}c$ are kept constant ($t{:}c$ is fixed at 10%). The first function f_1 is in fact a smooth least-squares fitting to the coordinates of the mean of the SC(2) series of airfoils. Each of the four subsequent functions, $f_{2,\ldots,5}$, is a least-squares fitting to the residuals of the preceding fit and each airfoil of the SC(2) series, and can be added to f_1 with different weightings, $w_{2,\ldots,5}$. Figure 2 shows the effect of adding each function with a weighting of $w_i = -0.5$. Nonsensical airfoils are produced by adding individual functions with such large weightings, but a high degree of geometry control is achieved by combining the functions and optimizing their weightings. As seen in Fig. 2, the second function has the effect of varying the camber toward the rear of the airfoil and is used as our optimization variable, while the remaining three functions are kept constant at zero, that is, no further deviation from the mean NASA airfoil is added by these functions.

Constant lift is maintained at $C_L = 0.6$ by varying the angle of attack α of each airfoil. At low to moderate values of α, that is, before the onset of flow separation and stall, C_L varies linearly with α (see, for example, Anderson [19]). Therefore, to find the correct value of α, the first airfoil geometry is simulated at an initially guessed α_1, and at α_2 is determined by the result of the first simulation and an estimate of $\Delta C_L/\Delta\alpha$. A third simulation at α_3, found from a linear interpolation through (α_1, C_{L1}) and (α_2, C_{L2}) is sufficient to attain the desired lift and an accurate value of $\Delta C_L/\Delta\alpha$. Subsequent geometries are simulated using α_1 and C_{L1} from the closest previously simulated airfoil, in terms of the Euclidean distance between variables, such that, after the first few geometries, it is rarely necessary to proceed further than α_2 before the desired lift is met to within 1%.

Figure 1 displays 101 C_D values computed in this way as w_2 varies from −0.3 to 0.1. The general trend in the data is clear, but there are significant random fluctuations in the C_D data about this trend. The noise seen in Fig. 1 is in fact predominantly discretization error. These errors, which are caused by finite mesh resolution, manifest as noise because the error fluctuates across the design space due to perturbations in the mesh as small changes in the geometry occur (we remesh at each geometry perturbation).

Although the level of convergence of each iterative CFD simulation naturally has a significant impact on the value of the force coefficients obtained, the convergence only has an impact on the noise in the data if the simulations for different geometries converge at varying rates. In our experience, simulations converge largely in unison across a design space [20,21] and therefore produce negligible noise in comparison to discretization error.

Errors produced through the application of boundary conditions on the problem will be highly problem-dependent. Here we iterate

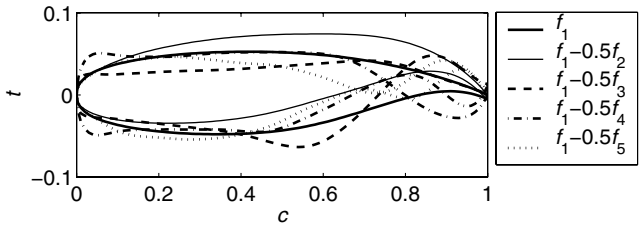


Fig. 2 Orthogonal basis function airfoil parameterization (axes not to scale).

toward a fixed lift by varying α. This results in very little noise in the C_D value, with most of the errors being transferred to α (noise in the input rather than in the output), and so the construction of the surrogate is not affected, because α is not an optimization variable.

III. Surrogate-Based Optimization

This paper is centered around surrogate-based optimization, and so we next present the method of kriging: an approximation method favored due to its ability to model complex functions and provide error estimates. Sacks et al. [4] popularized the use of kriging (outside of its birthplace in geostatistics) as a means to approximate the output of computer experiments. Here, equations for the kriging predictor and estimate of the potential prediction error are presented, with derivations given in the Appendix.

As with all surrogate-based methods, we start with a set of sample data, usually computed at a set of points in the domain of interest determined by a DOE procedure. The kriging approximation is (generally) built from a mean base term $\hat{\mu}$ [the circumflex denotes a maximum likelihood estimate (MLE)], plus n basis functions centered around the n sample points, $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, $\boldsymbol{x} \in \mathbb{R}^k$:

$$\hat{y}(\boldsymbol{x}_{n+1}) = \hat{\mu} + \sum_{i=1}^{n} b_i \psi_i(\| \boldsymbol{x}_{n+1} - \boldsymbol{x}_i \|) \tag{1}$$

where the basis functions ψ_i are given by the column vector:

$$\boldsymbol{\psi} = \begin{pmatrix} \mathrm{Corr}[Y(\boldsymbol{x}_{n+1}), Y(\boldsymbol{x}_1)] \\ \vdots \\ \mathrm{Corr}[Y(\boldsymbol{x}_{n+1}), Y(\boldsymbol{x}_n)] \end{pmatrix} = \begin{pmatrix} \exp\left[-\sum_{j=1}^{k} \hat{\theta}_j (\| x_{n+1,j} - x_{1,j} \|)^{\hat{p}_j}\right] \\ \vdots \\ \exp\left[-\sum_{j=1}^{k} \hat{\theta}_j (\| x_{n+1,j} - x_{n,j} \|)^{\hat{p}_j}\right] \end{pmatrix} \tag{2}$$

[the correlation between a random variable $Y(\boldsymbol{x})$ at the point to be predicted $(\boldsymbol{x}_{n+1})$ and at the sample data points $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$]. The hyper-parameter p_j can be thought of as determining the smoothness of the function approximation. In geostatistical models, where kriging originated, erratic local behavior may benefit from the use of $\hat{p}_j \in [0, 1]$ to allow for erratic responses, but here the modeling of engineering functions implies that there will not be any singularities and the use of $p_j = 2$ means that the basis function is infinitely differentiable through a sample point, when $\| \boldsymbol{x}_{n+1} - \boldsymbol{x}_i \| = 0$. With $p_j = 2$, the basis function is a Gaussian kernel with variance $1/\hat{\theta}_j$. Therefore, $\hat{\boldsymbol{\theta}}$ can be thought of as determining how quickly the function changes as $\boldsymbol{x}_{n+1}$ moves away from $\boldsymbol{x}_i$, with high and low $\hat{\theta}_j$ indicating an active or inactive function, respectively. It is usual practice to use a constant $\hat{\theta}_j$ for all dimensions in $\boldsymbol{x}$, but the use of variable $\hat{\theta}_j$ gives a non axisymmetric basis, allowing for varying impacts of each variable of the design space. In essence, the variance $1/\hat{\theta}_j$ is used to normalize the distance $\| x_{n+1,j} - x_{i,j} \|$ to give equal activity across each dimension [7].

The constants b_i are given by the column vector:

$$\boldsymbol{b} = \mathbf{R}^{-1}(\boldsymbol{y} - \mathbf{1}\hat{\mu})$$

where $\mathbf{R}$ is an $n \times n$ symmetric matrix of correlations between the sample data, $\boldsymbol{y}$ is a column vector of the sample data:

$$[y(\boldsymbol{x}_1), \ldots, y(\boldsymbol{x}_n)]^T$$

$\mathbf{1}$ is an $n \times 1$ column vector of ones, and the MLE of the mean is given by

$$\hat{\mu} = \mathbf{1}^T \mathbf{R}^{-1} \boldsymbol{y} / \mathbf{1}^T \mathbf{R}^{-1} \mathbf{1} \tag{3}$$

In addition to computing an initial set of experiments and fitting an approximation to the data, the surface is usually refined with additional data (update or infill points) to improve accuracy in the area of the optimum and confirm the objective function values predicted by the approximation. After each update the kriging model is rebuilt with hyper-parameters optimized for the augmented data set. An obvious way of refining the surface is to compute a new simulation at the predicted optimum. The approximation is then rebuilt and new optimum points are added until the predicted optimum agrees with the update simulation to a specified tolerance. The optimization may, however, become trapped at local optima when searching multimodal functions (see Jones [9] for an excellent review of the pitfalls of various update criteria). An update strategy must allow for the prediction being just that: a prediction. The kriging error is related to how robust our MLE of the predictor is and is given by

$$\hat{s}^2(\boldsymbol{x}_{n+1}) = \hat{\sigma}^2 (1 - \boldsymbol{\psi}^T \mathbf{R}^{-1} \boldsymbol{\psi}) \tag{4}$$

Where

$$\hat{\sigma}^2 = (\boldsymbol{y} - \mathbf{1}\hat{\mu})^T \mathbf{R}^{-1} (\boldsymbol{y} - \mathbf{1}\hat{\mu}) / n \tag{5}$$

(see the Appendix for derivations). (An extra term in the error,

$$\hat{\sigma}^2 \frac{(1 - \mathbf{1}^T \mathbf{R}^{-1} \boldsymbol{\psi})^2}{\mathbf{1}^T \mathbf{R}^{-1} \mathbf{1}} \ll 1$$

is attributed to the error in the estimate of $\hat{\mu}$, and is neglected here.) Equation (4) has the intuitive property that the error is zero at a sample point, because if $\boldsymbol{\psi}$ is the ith column of $\mathbf{R}$, then

$$\boldsymbol{\psi}^T \mathbf{R}^{-1} \boldsymbol{\psi} = \boldsymbol{\psi}(\boldsymbol{x}_i - \boldsymbol{x}_i) = 1$$

Positioning updates based on the error alone (i.e., maximizing $\hat{s}^2$) will, of course, lead to a completely global search, although the eventual location of a global optimum is guaranteed, because the sampling will be dense. Here we employ an infill criterion that balances local exploitation of $\hat{y}$ and global exploration using $\hat{s}^2$ by maximizing the expectation of improving upon the current best solution.

Kriging treats the value of the function at $\boldsymbol{x}$ as if it were the realization of a stochastic process $Y(\boldsymbol{x})$, with a probability density function:

$$\frac{1}{\sqrt{2\pi}\hat{s}(\boldsymbol{x})} \exp -\frac{1}{2}\left(\frac{Y(\boldsymbol{x}) - \hat{y}(\boldsymbol{x})}{\hat{s}(\boldsymbol{x})}\right)^2$$

with the mean given by the predictor $\hat{y}(\boldsymbol{x})$ [Eq. (1)] and variance $\hat{s}^2$ [Eq. (4)]. This allows us to model our uncertainty about the predictions we make. The most plausible value at $\boldsymbol{x}$ is $\hat{y}(\boldsymbol{x})$, with the probability decreasing as $Y(\boldsymbol{x})$ moves away from $\hat{y}(\boldsymbol{x})$. Because there is uncertainty in the value of $\hat{y}(\boldsymbol{x})$, we can calculate the expectation of it being an improvement, $I = f_{\min} - Y(\boldsymbol{x})$, on the best value calculated so far:

$$E[I(\boldsymbol{x})] = \int_{-\infty}^{\infty} \max\,(f_{\min} - Y(\boldsymbol{x}), 0)\phi(Y(\boldsymbol{x}))\mathrm{d}Y = \begin{cases} [f_{\min} - \hat{y}(\boldsymbol{x})]\boldsymbol{\Phi}\left(\frac{f_{\min} - \hat{y}(\boldsymbol{x})}{\hat{s}(\boldsymbol{x})}\right) + \hat{s}\phi\left(\frac{f_{\min} - \hat{y}(\boldsymbol{x})}{\hat{s}(\boldsymbol{x})}\right) & \text{if } \hat{s} > 0 \\ 0 & \text{if } \hat{s} = 0 \end{cases} \tag{6}$$

where $\boldsymbol{\Phi}(.)$ and $\phi(.)$ are the normal cumulative distribution function and probability density function, respectively. $f_{\min} - \hat{y}(\boldsymbol{x})$ should be replaced by $\hat{y}(\boldsymbol{x}) - f_{\max}$ for a maximization problem, but in practice it is easier to take the negative of the data so that all problems can be treated as minimizations (we consider minimization for the remainder of this paper). Note that $E[I(\boldsymbol{x})] = 0$ when $\hat{s} = 0$ so that there is no expectation of improvement at a point that has already been sampled and therefore no possibility of resampling, which is a necessary characteristic of an updating criterion when using deterministic computer experiments and guarantees global convergence. Without the possibility of resampling, as the number updates based on the maximum $E[I(\boldsymbol{x})]$ tends to infinity, the design space will

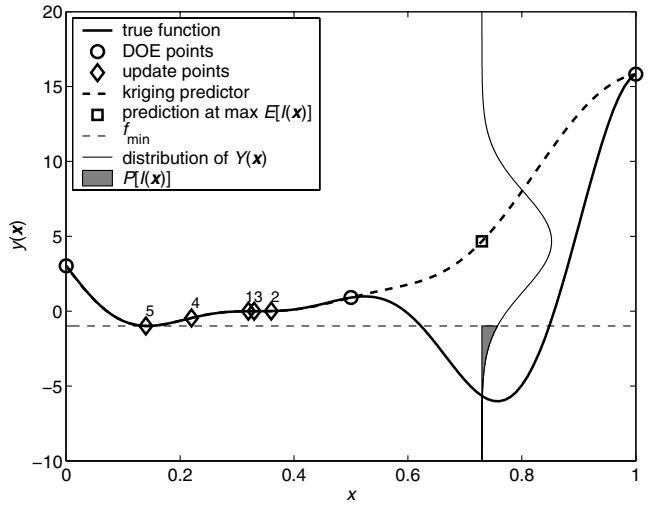


Fig. 3 Maximum $E[I(x)]$ updates to a test function with the maximum $E[I(x)]$ after the fifth update shown as the moment of the area under $Y(x)$, which is less than $f_{\min}$.

become densely populated and so the global optimum will be found. This is the basis of Locatelli's proof of convergence [15].

The meaning and use of $E[I(\boldsymbol{x})]$ is visually displayed in Fig. 3, which shows the progress of the optimization of a deceptive test function:

$$y(x) = (6x - 2)^2 \sin(12x - 4)$$

An initial DOE of three points is updated at $\max(E[I(\boldsymbol{x})])$. The situation is shown after five updates, the first three of which isolated one local optima (where a $\min[\hat{y}(\boldsymbol{x})]$ search would have stalled), with the fourth and fifth finding a second local minima. The maximum $E[I(\boldsymbol{x})]$ is now at $x = 0.73$. The distribution of $Y(\boldsymbol{x})$ is plotted (note that the mean is the kriging predictor) and from Eq. (6) we see that $E[I(\boldsymbol{x})]$ is given by the moment of the area (which is the probability of improvement $P[I(\boldsymbol{x})]$) of this distribution below the best value found so far. $Y(\boldsymbol{x})$ may take values below $f_{\min}$ due to the sparsity of data in this region leading to a high $\hat{s}^2$. After a sixth update at this location the optimization will quickly find the global minimum of the function.

Note that although here we concentrate on using $E[I(\boldsymbol{x})]$ as our update criterion, the methods presented will be equally applicable to other criteria that are based on error estimates, for example, probability of improvement.

IV. Optimizing Noisy Functions

A. Interpolation

We have already discussed the benefits of basis function interpolation methods (and, in particular, kriging) over polynomial regression models. However, such methods are prone to failure when dealing with noisy data. The smooth continuous approximation function is unable to fit the discontinuous numerical solution to the (presumably) smooth engineering function. Problems do not arise when data is sparse: the approximation can accommodate small perturbations in the data with a smooth function. However, as a search and update strategy converges on an optimum, the data becomes more dense and an interpolating model may predict erroneous results. Figure 4 shows the progress of an update strategy using kriging interpolation to model the C_D of the one-variable airfoil optimization.

The left-hand column of plots shows the kriging prediction of the function based on the sample data, whereas the right-hand column shows the expected improvement in the prediction. The first three updates, each one shown as a diamond in the respective plot, follow a logical progression toward the minimum drag design (shown as a cross), but the fourth update produces an erratic surface and high expected improvements in areas of poor designs. The true extent of how erratic the interpolating prediction has become is seen in Fig. 5.

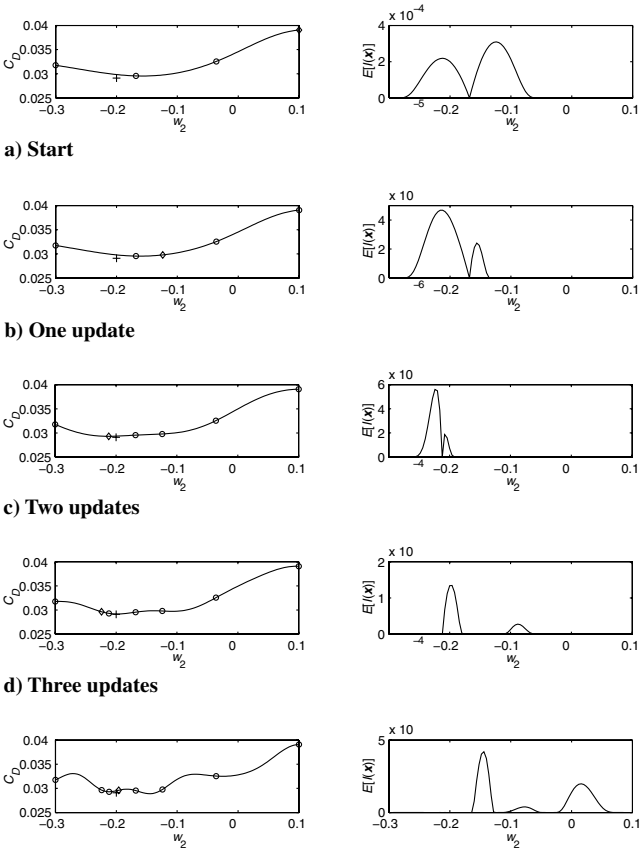


Fig. 4 Updates using maximum $E[I(x)]$ of a kriging interpolation. The optimum C_D is depicted by a cross.

It is easy to see where the optimum lies in this one-variable problem but, should such a situation occur in a multidimensional design space, the optimization may continue to search in areas of poor designs based on high expectations of improvement and increasingly inaccurate kriging predictions.

B. Kriging Regression

The problem of approximating a noisy function is resolved by allowing the kriging model to regress the data. This is achieved by adding a regression constant (often termed a regularization constant) λ to the leading diagonal of the kriging correlation matrix $\mathbf{R}$ [22,23],[‖] that is, we now have $\mathbf{R} + \lambda\mathbf{I}$.[¶]

Without the regression constant, each point is given an exact correlation with itself, forcing the predictor to pass through the sample points. The regression constant λ is now optimized along with the other hyper-parameters to allow the predictor:

$$\hat{y}_r(\boldsymbol{x}_{n+1}) = \hat{\mu}_r + \boldsymbol{\psi}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}(\boldsymbol{y} - \mathbf{1}\hat{\mu}_r) \tag{7}$$

(subscript r denotes regression) where

$$\hat{\mu}_r = \frac{\mathbf{1}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}\boldsymbol{y}}{\mathbf{1}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}\mathbf{1}} \tag{8}$$

to deviate from the sample points in order to achieve an improved likelihood of the data. The effect is seen in Fig. 5 where the final plot of Fig. 4 is reproduced, but this time a kriging regression is also

[‖]The work of Tikhonov and Arsenin [23] on regularization actually predates that of Hoerl and Kennard [22] on ridge regression, but only became well known in the West after the publication of the referenced work.

[¶]This formulation works on the assumption that the noise is purely output-dependent. For input-dependent noise, a parameterized spatial correlation function would be employed.

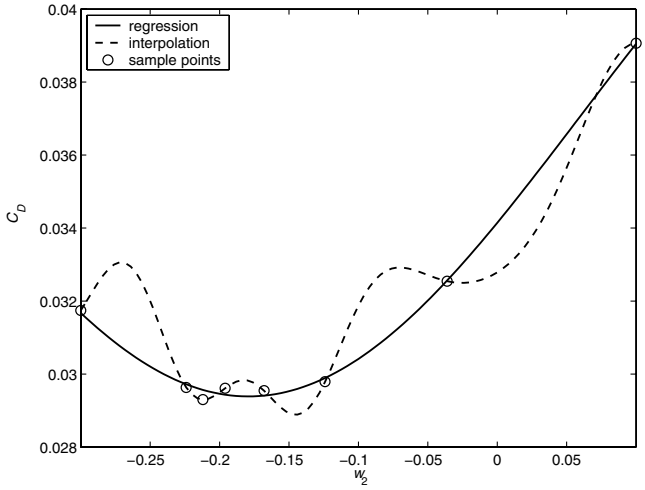


Fig. 5 Improved kriging approximation using regression compared with the final plot in Fig. 4.

shown and provides a more feasible approximation of the true function.

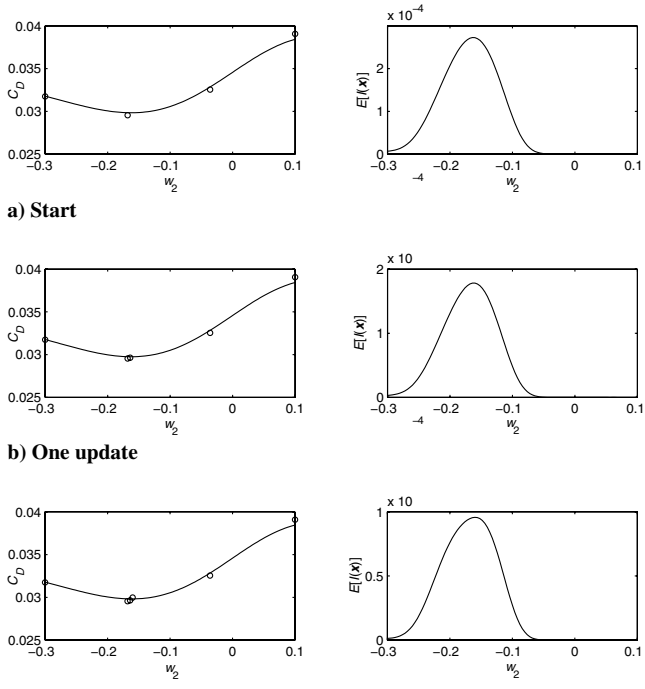
Optimization using a regressing approximation is, however, susceptible to further problems, not with the approximation model itself, but with the process of selecting update locations. Equation (6) dictates that the expected improvement is zero when the error [Eq. (4)] is zero (i.e., at the location of a sample point). However, when regressing, the error is now given by

$$\hat{s}_r^2(\boldsymbol{x}_{n+1}) = \hat{\sigma}_r^2[1 + \lambda - \boldsymbol{\psi}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}\boldsymbol{\psi}] \tag{9}$$

where

$$\hat{\sigma}_r^2 = (\boldsymbol{y} - \mathbf{1}\hat{\mu}_r)^T(\mathbf{R} + \lambda\mathbf{I})^{-1}(\boldsymbol{y} - \mathbf{1}\hat{\mu}_r)/n \tag{10}$$

(see the Appendix for derivations). This does not equal zero at a sample location, or indeed at any point. This results in the expectation of an improvement and therefore the possibility of maximizing the expected improvement at a previously sampled point. This is a plausible situation for a nondeterministic experiment,



a) Start

b) One update

c) Two updates

Fig. 6 Updates using maximum $E[I(x)]$ of a kriging regression.

because a repeated experiment may indeed lead to an improved function value. However, the error seen in a deterministic computer experiment is a repeatable error and recalculation will result in the same value and the optimization will become trapped at this point: repeated approximations will be identical, because no further data has been added to the model. Such a scenario is shown in Fig. 6. The starting point is the same as the previous interpolation example.

Figure 6 shows the initial prediction passing close to, but not through, the sample points, and so leading to a predicted error at these points. A high expected improvement is seen at the best sample point due to the low C_D and a predicted error at this location. The maximum of the expected improvement is in fact adjacent to the sample point and an update is applied here; the hyper-parameters are reoptimized including the new point and a new prediction is made. The process is repeated for a second update, but after this stage the expected improvement is at a maximum at the location of the first update point. The optimization is now stalled and cannot progress toward the optimum. Even if the maximum of the expected improvement does not occur at a sample point, it is seen from the first two updates that, before stalling, the optimization progresses very slowly when using this method of updating. The expected improvement is not diminishing because there is little change in the function or in the error, as all points are closely packed together. The incorrect approximation of the error also means that the sample points will not be dense and so global convergence cannot be guaranteed.

C. Reinterpolation

A more intuitively correct expected improvement is obtained if the error at the sample locations is assumed to be zero. Although there is error in all sample locations due to the noise in the data, by redefining the notion of the term error when using deterministic experiments to mean uncertainty in the result, this is a valid assumption.

Zero error at the sample locations is achieved by building an interpolating RSM through the values predicted by the kriging regression at the sample locations. The predictor of this reinterpolation is

$$\hat{y}(\boldsymbol{x}_{n+1}) = \hat{\mu} + \boldsymbol{\psi}^T\mathbf{R}^{-1}(\hat{\boldsymbol{y}}_r - \mathbf{1}\hat{\mu}) \tag{11}$$

where

$$\hat{\mu} = \frac{\mathbf{1}^T\mathbf{R}^{-1}\hat{\boldsymbol{y}}_r}{\mathbf{1}^T\mathbf{R}^{-1}\mathbf{1}} \tag{12}$$

and $\hat{\boldsymbol{y}}_r$ is a vector:

$$\hat{\boldsymbol{y}}_r = \begin{pmatrix} \hat{y}_r(\boldsymbol{x}_1), \\ \vdots \\ \hat{y}_r(\boldsymbol{x}_n) \end{pmatrix} \tag{13}$$

with elements $\hat{y}_r(\boldsymbol{x}_i)$ given by Eq. (7). $\mathbf{R}$ and $\boldsymbol{\psi}$ remain unchanged and so it is not necessary to reoptimize $\boldsymbol{\theta}$ and $\boldsymbol{p}$. An identical predictor to the original regression is obtained, but now the prediction of $\hat{s}^2$ is consistent with the data coming from a deterministic computer experiment. The sampling will now be dense and so the method will reach a global optimum.

To demonstrate that the predictors $\hat{y}_r(\boldsymbol{x})$ and $\hat{y}(\boldsymbol{x})$ are identical, it is first shown that $\hat{\mu} = \hat{\mu}_r$ by substituting Eq. (7) into Eq. (13) to give an expression for $\hat{\boldsymbol{y}}_r$, couched in terms of $\boldsymbol{y}$ and $\hat{\mu}_r$:

$$\hat{\boldsymbol{y}}_r = \mathbf{1}\hat{\mu}_r + \mathbf{R}(\mathbf{R} + \lambda\mathbf{I})^{-1}(\boldsymbol{y} - \mathbf{1}\hat{\mu}_r) \tag{14}$$

This is now substituted into Eq. (12) to give

$$\hat{\mu} = \frac{\mathbf{1}^T\mathbf{R}^{-1}\mathbf{1}\hat{\mu}_r + \mathbf{1}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}(\boldsymbol{y} - \mathbf{1}\hat{\mu}_r)}{\mathbf{1}^T\mathbf{R}^{-1}\mathbf{1}}$$
$$= \hat{\mu}_r + \frac{\mathbf{1}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}\boldsymbol{y} - \mathbf{1}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}\mathbf{1}\hat{\mu}_r}{\mathbf{1}^T\mathbf{R}^{-1}\mathbf{1}}$$

Noting, from Eq. (8), that $\mathbf{1}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}\mathbf{1}\hat{\mu}_r = \mathbf{1}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}\boldsymbol{y}$

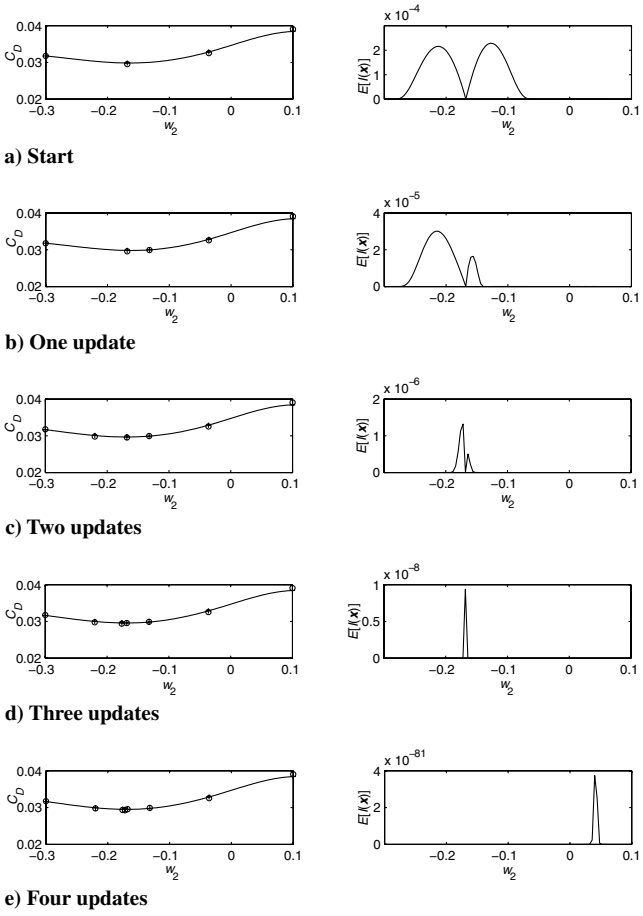


Fig. 7 Updates using maximum $E[I(x)]$ of a kriging reinterpolation.

$$\hat{\mu} = \hat{\mu}_r + \frac{\mathbf{1}^T(\mathbf{R}+\lambda\mathbf{I})^{-1}\mathbf{y}}{\mathbf{1}^T\mathbf{R}^{-1}\mathbf{1}} - \frac{\mathbf{1}^T(\mathbf{R}+\lambda\mathbf{I})^{-1}\mathbf{y}}{\mathbf{1}^T\mathbf{R}^{-1}\mathbf{1}} = \hat{\mu}_r$$

Now, substituting Eq. (14) into Eq. (11), and replacing $\hat{\mu}$ with $\hat{\mu}_r$, yields

$$\begin{aligned}\hat{y}(\boldsymbol{x}_{n+1}) &= \hat{\mu}_r + \boldsymbol{\psi}^T\mathbf{R}^{-1}(\mathbf{1}\hat{\mu}_r + \mathbf{R}(\mathbf{R}+\lambda\mathbf{I})^{-1}(\hat{\mathbf{y}} - \mathbf{1}\hat{\mu}_r) - \mathbf{1}\hat{\mu}_r)\\ &= \hat{\mu}_r + \boldsymbol{\psi}^T\mathbf{R}^{-1}\mathbf{1}\hat{\mu}_r + \boldsymbol{\psi}^T(\mathbf{R}+\lambda\mathbf{I})^{-1}(\hat{\mathbf{y}} - \mathbf{1}\hat{\mu}_r) - \boldsymbol{\psi}^T\mathbf{R}^{-1}\mathbf{1}\hat{\mu}_r\\ &= \hat{\mu}_r + \boldsymbol{\psi}^T(\mathbf{R}+\lambda\mathbf{I})^{-1}(\mathbf{y} - \mathbf{1}\hat{\mu}_r) = \hat{y}_r(\boldsymbol{x}_{n+1})\end{aligned}$$

[from Eq. (7)]. Thus, the two predictors given by Eqs. (7) and (11) are indeed identical. As such, the regression model may be used as a prediction and the reinterpolation used solely to calculate the error. Substituting Eq. (14) into the MLE of σ^2 [Eq. (5)] gives an expression for our estimate of σ^2 for the reinterpolation:

$$\hat{\sigma}^2_{\text{ri}} = \frac{(\mathbf{y}-\mathbf{1}\hat{\mu})^T(\mathbf{R}+\lambda\mathbf{I})^{-1}\mathbf{R}(\mathbf{R}+\lambda\mathbf{I})^{-1}(\mathbf{y}-\mathbf{1}\hat{\mu})}{n} \qquad (15)$$

In our equation for the interpolating kriging error [Eq. (4)], only $\hat{\sigma}^2$ depends on the predictor. We can therefore obtain our reinterpolation error estimate from Eq. (4) simply by replacing $\hat{\sigma}^2$ with $\hat{\sigma}^2_{\text{ri}}$. The error estimated using this expression reduces to zero at sample points, but uses a lower estimate of σ^2 reflecting the uncertainty in predicting the underlying trend of the data rather than the overall uncertainty that includes the noise. Note also that when there is no regression, $\lambda = 0$ and Eq. (15) reduces to Eq. (5).

An update strategy based on this method of reinterpolation is shown in Fig. 7. The initial prediction is identical to that in Fig. 6, but the expected improvement is based on the same prediction interpolating the points shown as crosses. The expected improvement diminishes to zero at all sample points and the regression model produces a smooth prediction of the function; guiding the search toward the optimum, despite the noise in the data. Note that, contrary to the interpolating update example in Fig. 4, the expected improvement diminishes steadily throughout the optimization (note that the scale of $E[I(x)]$ changes from plot to plot).

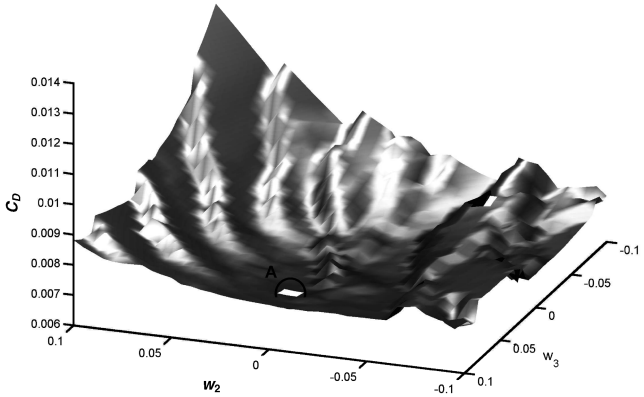


Fig. 8 True C_D response from VGK.

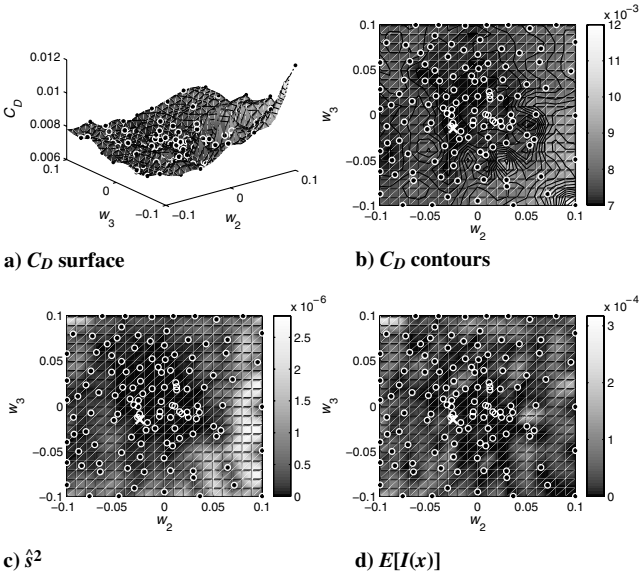


Fig. 9 Situation after 123 maximum $E[I(x)]$ updates using kriging interpolation.

V. Example Problem 2

The airfoil optimization problem is now extended to two variables, w_1 and w_2, and flow simulations are performed using the viscous Garabedian and Korn (VGK) code [24]: a full potential code with an empirical drag correction. With each calculation taking only a few seconds, it is possible to build a detailed map of the objective function that is to be optimized using a kriging model. This true response surface is shown in Fig. 8, where it is seen that the analysis shows a more regular pattern of noise than the Fluent solution. While the Fluent solution shows more random noise associated with the rebuilding of an unstructured mesh for each simulation, the VGK C_D map shows smooth contours punctuated by sudden changes as the structured mesh is altered to accommodate each shape design. Although the characteristics of the noise are different, the optimization still faces a similar problem: that of sudden discontinuities in the function, which do not represent the true performance of the design.

The blank portions of the surface, for example, region A, indicate failed simulations. Of the 1681 airfoil designs simulated to build Fig. 8, only three failed to converge. Despite the low failure rate, it is nevertheless imperative that an optimization process should be able

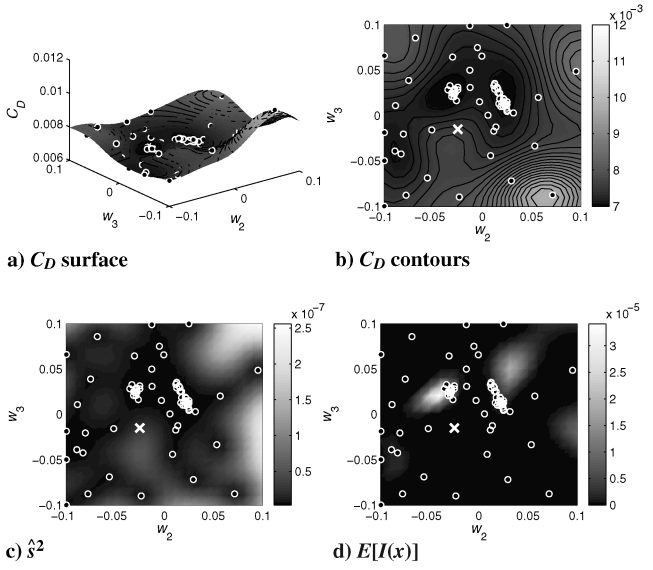


Fig. 10 Stalled maximum $E[I(x)]$ updates using kriging regression.

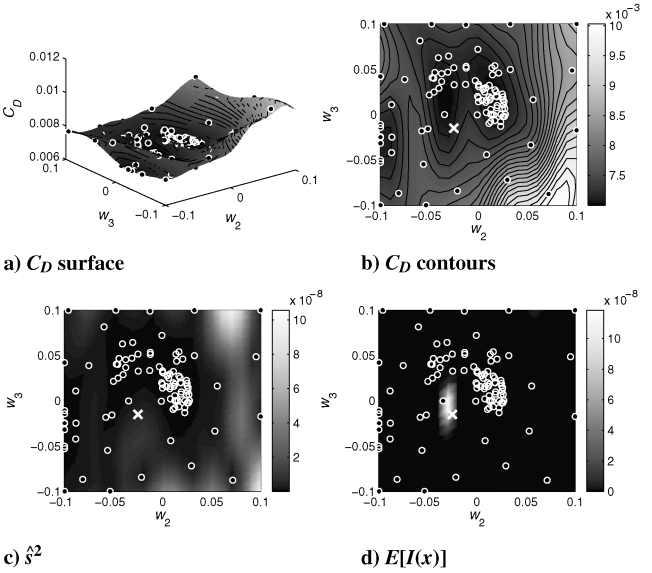


Fig. 11 Situation after 102 maximum $E[I(x)]$ updates using kriging reinterpolation.

to cope with a failed simulation. A small percentage of failures in the initial DOE of simulations can be ignored and the RSM built regardless. A larger DOE may in fact be used to account for such occurrences. Failures in the update process cannot be ignored though, because this would stall the optimization process: with no update, the RSM and expected improvement function remain unchanged. We circumvent the problem of failed simulations by imputing data at these points [25]. Imputations take the value of the predictor plus the mean squared error ($\hat{y}(\boldsymbol{x}) + \hat{s}^2$). This effectively assigns a statistical upper bound to the function to reduce the expectation of improvement, thus preventing the optimization from returning to these inadmissable design parameters, while retaining the property of global convergence in all other areas. The value of the imputations is renewed with a new prediction at each stage of the update process. This prevents distortion of the surface as the accuracy of the prediction is enhanced by successful update points.

We begin, as before, by using a kriging interpolation to model the C_D of the airfoil. The initial DOE comprises ten points, to which further points are augmented using the maximum expected improvement criterion until the optimum C_D of 0.0069 (found from the 41×41 evaluations in Fig. 8) is met to within 0.1 drag counts. This criterion is reached after 123 function evaluations. The resulting objective function, error, and $E[I(\boldsymbol{x})]$ surfaces are displayed in Fig. 9, with data points shown by white encircled black dots and the optimum depicted by a white cross. Note how the design space has been sampled extensively due to the highly multimodal nature of the $E[I(\boldsymbol{x})]$ surface, despite the simple underlying nature of the function. In a high dimensional optimization, such a scenario makes it extremely difficult to determine whether the search is converging on an optimum, because neither the position of updates nor $E[I(\boldsymbol{x})]$ are converging.

Figure 10 shows the result of applying updates based on a kriging regression, with the error defined by Eq. (9) (i.e., without reinterpolation). The search finds two basins of attraction (note the clustering of points in two regions), neither of which contain the global optimum, and stalls in one of these regions after 82 function evaluations with the best C_D being 0.00696. $E[I(\boldsymbol{x})]$ has a well-defined maximum at the cluster of points on the left, due to the strong likelihood of the data, however, this maximum occurs where there is already data present (to double precision accuracy).

The improvements afforded by using Eq. (15) to calculate the variance are seen in Fig. 11. We have a sensible prediction (compared with the true function), with a large number of the update simulations located in the basins of attraction found by the regression-based optimization. But, because the new error estimate returns to zero at the sample points, the search has been able to escape these regions and a global search has been performed. Note the greater spread of points across optimal regions, the extra points positioned at the extremities of the design space, and the all-important final point that locates the basin of attraction containing the global optimum, found after 102 function evaluations (the point just above and to the left of the optimum). Figure 11d shows there is now a well-defined maximum $E[I(\boldsymbol{x})]$ around the global optimum and that this is the only area where there is a visible $E[I(\boldsymbol{x})]$. The designer can be confident that the search is converging, due to convergence of the maximum $E[I(\boldsymbol{x})]$ to 10^{-7}, three orders of magnitude lower than the interpolation-based search.

Although the preceding figures give a helpful insight into the nature of the three kriging models being discussed, Table 1 presents a numerical comparison. Each method has again been run from a ten-point DOE until the C_D drops to less than 0.00691. To avoid the possibility of the results being DOE-dependent, averages are shown for ten optimizations with ten different Latin hypercube initial DOEs. It is seen from the table that, although interpolation reaches the target C_D in all cases, it would be hard to determine that an optimum had been found in an unfamiliar problem due to the instability of the correlation matrix producing poor error estimates and therefore

Table 1 Numerical comparison of interpolation, regression, and reinterpolation.

	Number of function evaluations		max $E[I(\boldsymbol{x})]$ at termination	
	mean	std	mean	std
Interpolation	58.7	42.2	5.25×10^{3}[a]	1.06×10^{4}
Regression	n/a[b]	n/a[b]	4.46×10^{-5}	3.89×10^{-5}
Reinterpolation	47.2	42.3	1.70×10^{-5}	3.09×10^{-5}

[a]The high expected improvement is due to $\mathbf{R}$ becoming close to singular when $\boldsymbol{\theta}$ increases to improve the likelihood of the interpolated noisy data. The result is high error estimates and a near random search.
[b]No average is available due to five optimizations stalling when a duplicate update point was selected. The average optimum is 0.0693 after an average of 41.1 evaluations with a standard deviation of 21.1.

high expectations. As expected, the regression model fails to find the optimum (in half the optimizations performed) after stalling when the maximum $E[I(\boldsymbol{x})]$ occurred at a previously sampled point. Furthermore, this maximum is low enough for us to have some (unfounded) confidence that a good optimum has been found. The reinterpolation model successfully finds the optimum for all ten runs, with correlations that provide good predictions, low error estimates, and therefore more appropriate expectations.

VI. Conclusion

The use of a regularization constant can provide improved surrogate models when noise exists in the sample data. However, this approach causes difficulties when performing global optimization using update criteria based on error estimates (e.g., expected improvement): regression alone can lead to a stalled update process with no guarantee of global convergence. By using the reinterpolation method suggested here, the global convergence of the maximum expected improvement criterion can be preserved while maintaining the benefits of regression. This leads to a significantly faster and more robust search procedure when dealing with computer experiments that are subject to computational noise.

Appendix

Here we derive the equations for the kriging predictor and its mean squared error. A classical derivation of the interpolating predictor and error is given by Schonlau [26], but we choose to follow the perhaps more intuitive method of Jones [9], which we modify to take account of regression. The derivations use the regression correlation matrix of $\mathbf{R} + \lambda\mathbf{I}$ and the interpolation equations are obtained simply by setting $\lambda = 0$. The subscripts r, used in the main body of the paper to distinguish between regression and interpolation, are omitted in the derivation.

We represent our uncertainty about the functions value at n sample points using the random vector:

$$\boldsymbol{Y} = \begin{pmatrix} Y(\boldsymbol{x}_1) \\ \vdots \\ Y(\boldsymbol{x}_n) \end{pmatrix}$$

with mean $\mathbf{1}\mu$ and covariance matrix $\mathrm{Cov}(\boldsymbol{Y}) = \sigma^2(\mathbf{R} + \lambda\mathbf{I})$.

We estimate μ, σ^2, $\boldsymbol{\theta}$, and $\boldsymbol{p}$ by maximizing the likelihood of the observed data $\boldsymbol{y}$, with the likelihood given by

$$\frac{1}{(2\pi)^{\frac{n}{2}}(\sigma^2)^{\frac{n}{2}}|\mathbf{R} + \lambda\mathbf{I}|^{\frac{1}{2}}} \exp\left(\frac{-(\boldsymbol{y} - \mathbf{1}\mu)^T(\mathbf{R} + \lambda\mathbf{I})^{-1}(\boldsymbol{y} - \mathbf{1}\mu)}{2\sigma^2}\right)$$

The maximization problem is simplified by taking the natural logarithm and ignoring constant terms to give

$$-\frac{n}{2}\ln(\sigma^2) - \frac{1}{2}\ln(|(\mathbf{R} + \lambda\mathbf{I})|) - \frac{(\boldsymbol{y} - \mathbf{1}\mu)^T(\mathbf{R} + \lambda\mathbf{I})^{-1}(\boldsymbol{y} - \mathbf{1}\mu)}{2\sigma^2} + \text{constant terms} \tag{A1}$$

By setting the first derivatives with respect to μ and σ^2 to zero, and solving, we obtain our MLEs:

$$\hat{\mu} = \frac{\mathbf{1}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}\boldsymbol{y}}{\mathbf{1}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}\mathbf{1}} \tag{A2}$$

and

$$\hat{\sigma}^2 = \frac{(\boldsymbol{y} - \mathbf{1}\hat{\mu})^T(\mathbf{R} + \lambda\mathbf{I})^{-1}(\boldsymbol{y} - \mathbf{1}\hat{\mu})}{n} \tag{A3}$$

Substituting Eqs. (A2) and (A3) into Eq. (A1) we obtain what is known as the concentrated log likelihood:

$$-\frac{n}{2}\ln(\hat{\sigma}^2) - \frac{1}{2}\ln(|(\mathbf{R} + \lambda\mathbf{I})|)$$

and this is the function we maximize to find the hyper-parameters $\hat{\boldsymbol{\theta}}$, $\hat{\boldsymbol{p}}$, and λ (using search techniques such as a genetic algorithm; see, for example, Keane and Nair [27]).

With MLEs for μ, σ^2, $\boldsymbol{\theta}$, and $\boldsymbol{p}$ found (and fixed), we now find a MLE $\hat{y}(\boldsymbol{x}_{n+1})$ (our prediction at a new point) that has a correlation vector $\boldsymbol{\psi}$ with the observed data. We concatenate the observed data with the new point to give an augmented vector of

$$\tilde{\boldsymbol{y}} = [\boldsymbol{y}^T\ \hat{y}(\boldsymbol{x}_{n+1})]^T$$

and an augmented correlation matrix:

$$\tilde{\mathbf{R}} + \lambda\mathbf{I} = \begin{pmatrix} \mathbf{R} + \lambda\mathbf{I} & \boldsymbol{\psi} \\ \boldsymbol{\psi}^T & 1 + \lambda\mathbf{I} \end{pmatrix}$$

Looking back to Eq. (A1) it is seen that only the third term of the augmented log likelihood depends on $\hat{y}(\boldsymbol{x}_{n+1})$ and so the quantity to be maximized is

$$\frac{-(\tilde{\boldsymbol{y}} - \mathbf{1}\hat{\mu})^T(\tilde{\mathbf{R}} + \lambda\mathbf{I})^{-1}(\tilde{\boldsymbol{y}} - \mathbf{1}\hat{\mu})}{2\hat{\sigma}^2} \tag{A4}$$

The maximization problem is solved in the following way. substituting expressions for $\tilde{\boldsymbol{y}}$ and $\tilde{\mathbf{R}} + \lambda\mathbf{I}$ into Eq. (A4) yields

$$\frac{-\begin{pmatrix} \boldsymbol{y} - \mathbf{1}\hat{\mu} \\ \hat{y}(\boldsymbol{x}_{n+1}) - \hat{\mu} \end{pmatrix}\begin{pmatrix} \mathbf{R} + \lambda\mathbf{I} & \boldsymbol{\psi} \\ \boldsymbol{\psi}^T & 1 + \lambda\mathbf{I} \end{pmatrix}^{-1}\begin{pmatrix} \boldsymbol{y} - \mathbf{1}\hat{\mu} \\ \hat{y}(\boldsymbol{x}_{n+1}) - \hat{\mu} \end{pmatrix}}{2\hat{\sigma}^2} \tag{A5}$$

Using the partitioned inverse formula [28], the inverse augmented correlation matrix can be expressed as

$$\begin{pmatrix} \mathbf{R}_r^{-1} + \mathbf{R}_r^{-1}\boldsymbol{\psi}(1 + \lambda - \boldsymbol{\psi}^T\mathbf{R}_r^{-1}\boldsymbol{\psi})^{-1}\boldsymbol{\psi}^T\mathbf{R}_r^{-1}| & -\mathbf{R}_r^{-1}\boldsymbol{\psi}(1 + \lambda - \boldsymbol{\psi}^T\mathbf{R}_r^{-1}\boldsymbol{\psi})^{-1} \\ -(1 + \lambda - \boldsymbol{\psi}^T\mathbf{R}_r^{-1}\boldsymbol{\psi})^{-1}\boldsymbol{\psi}^T\mathbf{R}_r^{-1}| & (1 + \lambda - \boldsymbol{\psi}^T\mathbf{R}_r^{-1}\boldsymbol{\psi})^{-1} \end{pmatrix}$$

where $\mathbf{R}_r = \mathbf{R} + \lambda\mathbf{I}$, which, substituted into Eq. (A5) and ignoring terms without $\hat{y}(\boldsymbol{x}_{n+1})$, gives a new expression for the augmented log likelihood:

$$\left(\frac{-1}{2\hat{\sigma}^2(1 + \lambda - \boldsymbol{\psi}^T\mathbf{R}_r^{-1}\boldsymbol{\psi})}\right)(\hat{y}(\boldsymbol{x}_{n+1}) - \hat{\mu})^2 + \left(\frac{\boldsymbol{\psi}^T\mathbf{R}_r^{-1}(\boldsymbol{y} - \mathbf{1}\hat{\mu})}{\hat{\sigma}^2(1 + \lambda - \boldsymbol{\psi}^T\mathbf{R}_r^{-1}\boldsymbol{\psi})}\right)(\hat{y}(\boldsymbol{x}_{n+1}) - \hat{\mu}) \tag{A6}$$

This quadratic expression can be solved by setting its derivative with respect to $\hat{y}(\boldsymbol{x}_{n+1})$ equal to zero to give the value of $\hat{y}(\boldsymbol{x}_{n+1})$ that maximizes the augmented likelihood: the kriging predictor

$$\hat{y}(_{n+1}) = \hat{\mu} + \boldsymbol{\psi}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}(\boldsymbol{y} - \mathbf{1}\hat{\mu})$$

This is Eq. (1) with the basis functions ψ_i and constants b_i written explicitly.

As we noted when presenting Eq. (4), the error is related to how robust our MLE $\hat{y}(\boldsymbol{x}_{n+1})$ is. If the augmented likelihood falls off sharply as the value at $\boldsymbol{x}_{n+1}$ deviates from $\hat{y}(\boldsymbol{x}_{n+1})$, it follows that we

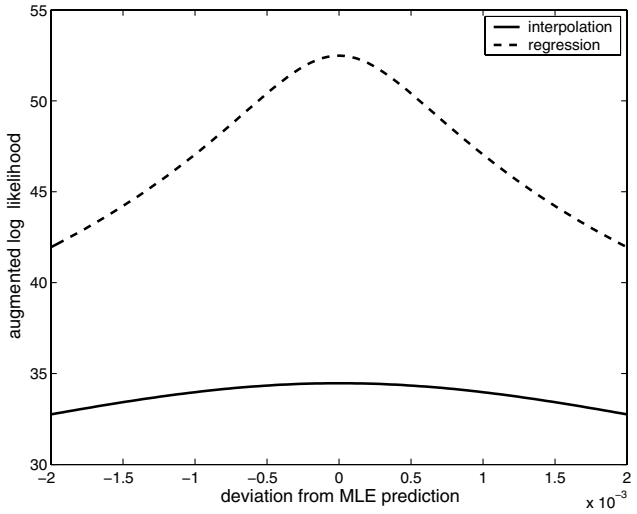


Fig. A1 Variation in the augmented log likelihood as the kriging prediction alters.

can be confident in our prediction. If different values give similar likelihoods, then the error in our MLE is likely to be higher. As an example, the augmented likelihoods of varying values for the interpolating and regressing prediction in Fig. 5 at $\boldsymbol{x} = -0.1$ are shown in Fig. A1. The interpolating model in Fig. 5 is highly multimodal (with a high $\hat{\boldsymbol{\theta}}$) and we intuitively expect the error to be high. As expected, Fig. A1 shows there is a poor likelihood of $\hat{y}(\boldsymbol{x}_{n+1})$, with very little change in this likelihood as the prediction varies on the x axis. The regressing model, on the other hand, is seen to model the data well. There is a high likelihood of $\hat{y}(\boldsymbol{x}_{n+1})$, which drops off sharply as we move away from the smooth trend seen in Fig. 5. From this line of thought, Jones [9] argues that the error is related to the inverse of the curvature of the augmented log likelihood. By taking the reciprocal of the second derivative with respect to $\hat{y}(\boldsymbol{x}_{n+1})$ of the augmented log likelihood [Eq. (A6)] we obtain our equation for the kriging error:

$$\hat{s}^2(\boldsymbol{x}_{n+1}) = \hat{\sigma}^2[1 + \lambda - \boldsymbol{\psi}^T(\mathbf{R} + \lambda\mathbf{I})^{-1}\boldsymbol{\psi}]$$

This equation is missing the

$$\hat{\sigma}^2 \frac{(1 - \mathbf{1}^T\mathbf{R}^{-1}\boldsymbol{\psi})^2}{\mathbf{1}^T\mathbf{R}^{-1}\mathbf{1}}$$

term of the classically derived formula (attributed to the error in the estimate of $\hat{\mu}$), but this term is so small it can safely be neglected.

Acknowledgements

This work has been supported by the Engineering and Physical Sciences Research Council: grant code GR/T19209/01. The authors also thank András Sóbester, Prasanth Nair, Nicola Hoyle, Thomas Barrett (University of Southampton), and Don Jones (General Motors) for their useful suggestions.

References

[1] Box, G. E. P., and Draper, N. R., *Empirical Model-Building and Response Surfaces*, Wiley Series in Probability and Mathematical Statistics, Wiley, New York, 1987.

[2] Matheron, G., "Principles of Geostatistics," *Economic Geology*, Vol. 58, No. 8, 1963, pp. 1246–1266.

[3] Krige, D. G., "A Statistical Approach to Some Basic Mine Valuation Problems on the Witwatersrand," *Journal of the Chemical, Metallurgical and Mining Society of South Africa*, Vol. 52, No. 6, 1951, pp. 119–139.

[4] Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H., "Design and Analysis of Computer Experiments," *Statistical Science*, Vol. 4, No. 4, 1989, pp. 409–423.

[5] Cressie, N. A. C., "Geostatistics," *The American Statistician*, Vol. 43, No. 4, 1989, pp. 197–202.

[6] Cressie, N. A. C., *Statistics for Spatial Data*, Probability and Mathematical Statistics, rev. ed., Wiley, New York, 1993.

[7] Jones, D. R., Schlonlau, M., and Welch, W. J., "Efficient Global Optimisation of Expensive Black-Box Functions," *Journal of Global Optimization*, Vol. 13, No. 4, 1998, pp. 455–492.

[8] Gutmann, H. M., "A Radial Basis Function Method for Global Optimization," *Journal of Global Optimization*, Vol. 19, No. 3, 2001, pp. 201–227.

[9] Jones, D. R., "A Taxonomy of Global Optimization Methods Based on Response Surfaces," *Journal of Global Optimization*, Vol. 21, No. 4, 2001, pp. 345–383.

[10] Sóbester, A., Leary, S. J., and Keane, A. J., "On the Design of Optimization Strategies Based on Global Response Surface Approximation Models," *Journal of Global Optimization*, Vol. 33, No. 1, 2005, pp. 31–59.

[11] Narducci, R., Grossman, B., Valorani, M., Dadone, A., and Haftka, R. T., "Optimization Methods for Non-Smooth or Noisy Objective Functions in Fluid Design Problems," AIAA Paper 1995-1648, June 1995.

[12] Giunta, A. A., "Aircraft Multidisciplinary Design Optimization Using Design of Experiments Theory and Response Surface Modeling Methods," Ph.D. Dissertation, Virginia Polytechnic Inst. and State Univ., Blacksburg, VA, May 1997.

[13] Papila, M., and Haftka, R. T., "Response Surface Approximations: Noise, Error Repair, and Modeling Errors," *AIAA Journal*, Vol. 38, No. 12, 2000, pp. 2336–2343.

[14] Kim, H., Papila, M., Mason, W., Haftka, R. T., Watson, L. T., and Grossman, B., "Detection and Repair of Poorly Converged Optimization Runs," *AIAA Journal*, Vol. 39, No. 12, 2001, pp. 2242–2249.

[15] Locatelli, M., "Bayesian Algorithms for One-Dimensional Global Optimization," *Journal of Global Optimization*, Vol. 10, No. 1, 1997, pp. 57–76.

[16] Fluent, *Fluent User Guide*, Fluent, Lebanon, NH, 2003.

[17] Robinson, G. M., and Keane, A. J., "Concise Orthogonal Representation of Supercritical Aerofoils," *Journal of Aircraft*, Vol. 38, No. 3, 2001, pp. 580–583.

[18] Harris, C. D., "NASA Supercritical Airfoils–A Matrix of Family-Related Airfoils," NASA TP 2969, March 1990.

[19] Anderson, J. D., *Introduction to Flight*, Aerospace Science Series, 3rd ed., McGraw–Hill, Singapore, 1989.

[20] Forrester, A. I. J., "Efficient Global Optimisation Using Expensive CFD Simulations," Ph.D. Dissertation, Univ. of Southampton, Southampton, England, Nov. 2004.

[21] Forrester, A. I. J., Bressloff, N. W., and Keane, A. J., "Optimization Using Surrogate Models and Partially Converged Computational Fluid Dynamics Simulations," *Proceedings of the Royal Society*, Series A, Vol. 462, No. 2071, 2006, pp. 2177–2204.

[22] Hoerl, A. E., and Kennard, R. W., "Ridge Regression: Biased Estimation for Nonorthogonal Problems," *Technometrics*, Vol. 12, No. 1, 1970, pp. 55–67.

[23] Tikhonov, A. N., and Arsenin, V. Y., *Solutions of Ill-Posed Problems*, Winston, Washington, D.C., 1977.

[24] Freestone, M. M., "VGK Method for Two-Dimensional Aerofoil Sections," Engineering Sciences Data Unit, TR 96028, Nov. 1996.

[25] Forrester, A. I. J., Sóbester, A., and Keane, A. J., "Optimization with Missing Data," *Proceedings of the Royal Society of London*, Series A, Vol. 462, No. 2067, 2006, pp. 935–945.

[26] Schonlau, M., "Computer Experiments and Global Optimization," Ph.D. Dissertation, Univ. of Waterloo, Waterloo, Ontario, Canada, 1997.

[27] Keane, A. J., and Nair, P. B., *Computational Approaches for Aerospace Design: The Pursuit of Excellence*, Wiley, New York, 2005.

[28] Theil, H., *Principles of Econometrics*, Wiley, New York, 1971.

N. Alexandrov
Associate Editor