Proceedings of IDETC/CIE 2007
ASME 2007 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
September 4-7, 2007, Las Vegas, Nevada, USA

DETC2007-34662

USING MAXIMUM LIKELIHOOD ESTIMATION TO ESTIMATE KRIGING MODEL PARAMETERS

Jay D. Martin*
Research Associate
Applied Research Laboratory
N. Atherton St.
State College, Pennsylvania, 16804-0030
Email: jdm111@psu.edu

ABSTRACT

A kriging model can be used as a surrogate to a more computationally expensive model or simulation. It is capable of providing a continuous mathematical relationship that can interpolate a set of observations. One of the major issues with using kriging models is the potentially computationally expensive process of estimating the best model parameters. One of the most common methods used to estimate model parameters is Maximum Likelihood Estimation (MLE). MLE of kriging model parameters requires the use of numerical optimization of a continuous but possibly multi-modal log-likelihood function. This paper presents some enhancements to gradient-based methods to make them more computationally efficient and compares the potential reduction in computational burden. These enhancements include the development of the analytic gradient and Hessian for the log-likelihood equation of a kriging model that uses a Gaussian spatial correlation function. The suggested algorithm is very similar to the Scoring algorithm traditionally used in statistics, a Newton-Raphson gradient-based optimization method.

1 INTRODUCTION

The kriging model is seeing a significant increase in its application as a computationally efficient surrogate model in area of Multidisciplinary Design Optimization (MDO). One of the major drawbacks of further integration of the kriging model as a tool used for MDO has been the availability of off-the-shelf software or easy to implement algorithms. A second drawback has been the perception that estimating model parameters can be a very computationally expensive process. This paper presents kriging models in a clear manner that is easy to understand and implement. It presents algorithms that more efficiently estimate the kriging model parameters. The key element of this work is the derivation of the analytical gradient and Hessian for the log-likelihood equation when a Gaussian function is used to quantify the spatial correlation, the most common spatial correlation function used when approximating computationally expensive computer models [1].

This paper provides an introduction to the kriging model. The importance and meaning of the kriging model's form and different model parameters are clearly identified and described. This description of the kriging model also details the use of Maximum Likelihood Estimation (MLE) as a method for estimating the best kriging model parameters. MLE requires an optimization algorithm to maximize the logarithm of the likelihood equation and is often the most computationally expensive element of the successful creation and use of the kriging model as an approximation model.

There are two primary issues concerned with MLE optimization. The first is that using a gradient-based Newton-Raphson method on the log-likelihood equation can become very computationally expensive as more parameters (dimensions d) or observations n are added to the kriging model. This increased computational expense is the result of calculating both a gradient and a Hessian at each iteration of the optimization algorithm. Without an analytical representation of the gradient and Hessian of the log-likelihood equation, they are often approximated with either finite difference methods or Quasi-Newton methods. All of these approximation methods require the inversion of an $n \times n$ matrix (the correlation matrix) at each different value of the model parameters during the evaluation of the log-likelihood equation. The second issue is that, the log-likelihood equation may have multiple local maxima, resulting in the possible failure of a gradient-based method to determine the global maximum. To resolve both of these issues, it has become common to use a non-gradient based optimization algorithm such as

*Address all correspondence to this author.

simulated annealing [1]. This may be a poor solution since it: 1) still requires the inversion of the matrix at each iteration, 2) may require a large number of iterations to converge, and 3) requires that the user understands the impact of the tuning parameters associated with the optimization algorithm.

This paper presents a solution to reduce the computational expense of estimating the gradient and Hessian: a closed-form analytical solution to the gradient and Hessian of the log-likelihood equation when using a Gaussian spatial correlation function. Evaluating the gradient and Hessian at a fixed point only requires a single inversion to the correlation matrix. The equations can be represented quite succinctly on paper, but may be difficult to implement in code since they require the evaluation of three- and four-dimensional tensors. Some recommendations on evaluating the gradient and Hessian are given to assist in the implementation of the optimization algorithms.

As an application of these derived analytical forms, this paper compares the use of three different variations of gradient-based optimization algorithms to solve the MLE. These algorithms include the Newton-Raphson method, the traditional Scoring method, and simplified version of the Scoring method. All of the methods use the Levenburg-Marquardt [2] modification to improve convergence efficiency. All of these algorithms require the selection of starting values. The starting values can determine how efficiently the specific optimization method converges to the local optimum. A method is suggested that draws random samples of the starting values for the correlation distances from an exponential distribution. In order to provide an estimate of the global maximum, random starting values are used until the best value is returned three more times.

The paper is organized by first presenting some background on the kriging model: the history of its origins, the derivation of its form and equations, a brief description of Maximum Likelihood Estimation for kriging models, and a description of Newton-Raphson methods that can be used to optimize the log-likelihood equation. The next section provides a derivation of the analytical forms for the gradient and Hessian of the log-likelihood equation of the kriging model. Section 4 provides some more specific details on how to optimize the log-likelihood equations based upon the past experiences of the author. The last two sections compare the performance of three different methods to calculate or approximate the Hessian used in a gradient-based optimization algorithm and draw conclusions from the results of the comparisons.

2 BACKGROUND OF KRIGING

The kriging model's origins lie in the geostatistical work of Danae Krige [3]. His original work developed a model to more precisely quantify the spatial variability of ore-grades in a deposit based on spatially distributed core samples. The results of his variance estimating models were used to determine the most probable locations in which to mine the highest quality gold ore in a mine. The term "kriging" was coined by Matheron [4] to refer to the process of estimating the weights to apply to each observation when estimating an unobserved location that results in the minimal variance of that estimate. These weightings quantify the spatial correlation of the observations.

In the field of geostatistics, the most common method used to estimate model parameters, primarily those parameters that quantify the spatial correlation and variance, is a semivariogram of the observations. A semivariogram is a graphical representation of the average dissimilarity between the observed points as a function of the distance between them [5]. From a semivariogram it is possible to visually estimate the long range variance of the observations (i.e., the sill value or process variance as defined in the next section), the short range variance (i.e., the nugget-effect or the variance in the observation measurements), and the distance at which the variance (i.e., correlation) of the observations tend to move from the short range variance to the long range variance. The use of these graphical methods to select model parameters are embraced because they permit the geostatistician the opportunity to interpret the observations of the physical behavior and qualify the results with their expert knowledge.

In the field of statistics, a model of similar form was developed by Goldberger [6] at about the same time as Matheron. Goldberger's work extended the form of the generalized linear regression model to improve its efficiency, i.e. to reduce the prediction variance associated with the use of the model. A generalized linear regression model assumes that the residuals of the observations from the model are uncorrelated. In reality, these residuals are often spatially correlated, i.e. nearby residuals are correlated. Goldberger's work provided a means to quantify and include this spatial correlation in the predictive model.

2.1 Kriging Model Form

The mathematical form of a kriging model has two parts as shown in Eq. (1). The first part is a linear regression of the data with k regressors that model the drift of the process mean, the *trend*, over the domain. Most engineering applications of the kriging model use a constant trend model over the domain and rely upon the second part of the model to "pull" the response surface through the observed data by quantifying the correlation of nearby points [1].

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{k} \beta_i f_i(\mathbf{x}) + Z(\mathbf{x}) \tag{1}$$

The second part, $Z(\mathbf{x})$, is a model of a stationary Gaussian random process with zero mean and covariance

$$\mathrm{Cov}(Z(\mathbf{x}_1), Z(\mathbf{x}_2)) = \sigma^2 R(\mathbf{x}_1, \mathbf{x}_2). \tag{2}$$

The process variance, σ^2, is a scalar parameter of the spatial correlation function (SCF), $R(\mathbf{x}_1, \mathbf{x}_2)$.

There are two important aspects to choosing the model form within the constraints of Eqs. (1) and (2): 1) choice of the trend function regressors, $\mathbf{f}$, and 2) choice of the SCF. The most common choice of trend function regression, when using a kriging model as a deterministic approximation of a computer model, is a constant trend function, $\mathbf{f}(\mathbf{x}) = \{1\}$. This is due to the ability of the correlation function to effectively model the observed variations of the computer model's output over it's domain [7]. A nonconstant or non-trivial trend function can, in general, be used to improve the predictive capabilities and its efficiency, i.e. reduce the predicted variance across the model's domain [8].

The trend function's purpose in the kriging model is to remove the bias present in the observations, leaving residuals that have a zero mean as required by the second part of Eq. (1). Frequently in engineering design, a low fidelity model is available to estimate the more computationally expensive model observations. This lower fidelity model can then be used as the trend model, permitting the kriging model to calibrate the low fidelity model with the high fidelity results and provide a measurement of the uncertainty in the calibrated model. If a low fidelity model does not exist, then any prior knowledge on important elements to the form for the trend model should be included. If nothing is known of the observations being modeled, then a statistical test, based on the corrected Akaike Information Criterion can be used to determine the best parameters to include in the trend model [9].

The purpose of the SCF used in the second part of Eq. (1) is to quantify the observed correlations of the residuals. The are four SCFs that are commonly used with kriging models [10]: Gaussian, exponential, cubic spline and Matérn functions. All of these SCFs use a parameter to control the range of influence of nearby points. The exponential and Matérn functions have an additional parameter that controls the smoothness of the SCF. The Gaussian and exponential functions are special cases of the Matérn function. The Matérn function is suggested to be the best SCF due to its flexibility and asymptotic properties [11]. The Gaussian function is the most commonly used SCF in engineering design since it provides a relatively smooth surface, making it a better choice when used with gradient-based optimization algorithms [12]. It also requires the estimation of a single parameter for each dimension rather than the two required for the Matérn function. Often when creating kriging models in engineering design, there are an insufficient number of observations available to adequately choose an appropriate smoothness parameter for each input dimension. The *d*-dimensional Gaussian SCF [13,14] is defined by combining the Gaussian SCF and the product correlation rule to result in

$$R(\mathbf{x}_1,\mathbf{x}_2)=\prod_{i=1}^{d}\mathrm{e}^{-\left(\frac{\mathbf{x}_{2,\mathrm{i}}-\mathbf{x}_{1,\mathrm{i}}}{\theta_\mathrm{i}}\right)^2} \tag{3}$$

where $\mathbf{x}_{1,i}$ is the ith dimension of the vector $\mathbf{x}_1$ and θ_i is the correlation range parameter in the ith dimension. When calculating the multivariate correlation using the Gaussian SCF, it is more computationally efficient to add the univariate distance ratios and calculate the exp function only once rather than d times and multiplying the resulting values as indicated in Eq. (3).

Define the locations of a set of n observations of the computer model as $\mathbf{X}=\{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_n\}\subset\Omega$, where Ω is the set of all possible inputs to the model that result in an output, i.e., the domain of the computer model. The resulting outputs are $\mathbf{y}=\{y(\mathbf{x}_1),y(\mathbf{x}_2),\ldots,y(\mathbf{x}_n)\}$. Given these sampled outputs of the computer model, consider a linear estimator of the output,

$$\hat{y}(\mathbf{x})=\boldsymbol{\lambda}^{\mathrm{T}}(\mathbf{x})\mathbf{y}, \tag{4}$$

at any point $\mathbf{x}\in\Omega$. The kriging approach treats $\hat{y}(\mathbf{x})$ as a random function and finds the best linear unbiased predictor, $\boldsymbol{\lambda}^{\mathrm{T}}(\mathbf{x})\mathbf{y}$, which minimizes the mean square error of the prediction,

$$\mathrm{MSE}[\hat{y}(\mathbf{x})]=\mathrm{E}\left[\left(\boldsymbol{\lambda}^{\mathrm{T}}(\mathbf{x})\mathbf{y}-y(\mathbf{x})\right)^2\right], \tag{5}$$

subject to the unbiasedness constraint,

$$\mathrm{E}\left[\boldsymbol{\lambda}^{\mathrm{T}}(\mathbf{x})\mathbf{y}-y(\mathbf{x})\right]=0. \tag{6}$$

A few definitions are needed before the solution to the optimization problem of Eq. (5) subject to the constraint of Eq. (6) can be given. Firstly, a matrix $\mathbf{F}$ is constructed by evaluating the vector $\mathbf{f}(\mathbf{x})$ at each of the n known observations,

$$\mathbf{F}=\{\mathbf{f}(\mathbf{x_1}),\mathbf{f}(\mathbf{x_2}),\ldots,\mathbf{f}(\mathbf{x_n})\}^{\mathrm{T}}. \tag{7}$$

A vector $\mathbf{r}(\mathbf{x})$ that represents the correlation between an unknown point, $\mathbf{x}\in\Omega$, and the n known sample points is defined as:

$$\mathbf{r}(\mathbf{x})=\{R(\mathbf{x},\mathbf{x}_1),R(\mathbf{x},\mathbf{x}_2),\ldots,R(\mathbf{x},\mathbf{x}_n)\}^{\mathrm{T}}. \tag{8}$$

Given the set of observations of the computer model $\mathbf{X}$, the kth element, $k=1,\ldots,n$ of the vector function $\mathbf{r}(\mathbf{x})$ is $\mathbf{r}_k(\mathbf{x})$. The correlation matrix $\mathbf{R}$ quantifies the correlation of the observations to themselves. It a square $(n\times n)$ matrix that is symmetric about its diagonal. Its kth row (or column) is defined as

$$\mathbf{R}_k=\mathbf{r}(\mathbf{X}_k) \tag{9}$$

where $\mathbf{X}_k$ is the kth observation in $\mathbf{X}$. The covariance function $\mathbf{v}(\mathbf{x})$ quantifies the covariance of an unobserved location $\mathbf{x}\in\Omega$ to the set of observations $\mathbf{X}$. It scales the correlation function $\mathbf{r}(\mathbf{x})$ with the process variance σ^2 and is defined as

$$\mathbf{v}(\mathbf{x})=\sigma^2\mathbf{r}(\mathbf{x}) \tag{10}$$

Finally, the covariance matrix $\mathbf{V}$, like the covariance function $\mathbf{v}(\mathbf{x})$, is a scalar multiple of the correlation matrix $\mathbf{R}$ and is defined as

$$\mathbf{V}=\sigma^2\mathbf{R} \tag{11}$$

The estimate to $\boldsymbol{\lambda}(\mathbf{x})$ that solves the minimization problem of Eq. (5) subject to the unbiasedness constraint of Eq. (6) is

$$\hat{\boldsymbol{\lambda}}(\mathbf{x})=\mathbf{R}^{-1}\mathbf{F}\left(\mathbf{F}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{F}\right)^{-1}\mathbf{f}(\mathbf{x})+\mathbf{R}^{-1}\left[\mathbf{I}-\mathbf{F}\left(\mathbf{F}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{F}\right)^{-1}\mathbf{F}^{\mathrm{T}}\mathbf{R}^{-1}\right]\mathbf{r}(\mathbf{x}), \tag{12}$$

where $\mathbf{I}$ is the n-dimensional identity matrix. The best linear unbiased estimator (BLUE) of $\hat{y}(\mathbf{x})$ results from plugging the the estimate from Eq (12) into Eq. (4) [6]. The BLUE of $\hat{y}(\mathbf{x})$ is then given by

$$\hat{y}(\mathbf{x})=\mathbf{f}^{\mathrm{T}}(\mathbf{x})\hat{\boldsymbol{\beta}}+\mathbf{r}^{\mathrm{T}}(\mathbf{x})\mathbf{R}^{-1}\left(\mathbf{y}-\mathbf{F}\hat{\boldsymbol{\beta}}\right), \tag{13}$$

where the generalized least squares estimate of $\boldsymbol{\beta}$ is

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{F}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{F}\right)^{-1}\mathbf{F}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{y}. \tag{14}$$

The first component of Eq. (13) is the generalized least squares estimate of a point, $x \in \Omega$, given the correlation matrix, $\mathbf{R}$. The second component of Eq. (13) "pulls" the generalized least squares estimate through the observed data points, providing a deterministic response surface that interpolates all of the observations. The BLUE defined in Eq. (13) assumes the correlation parameters $\boldsymbol{\theta}$, used to define the spatial correlation of the observations, are known *a priori*. This is seldom the case and therefore the correlation parameters must also be estimated from the set of observations. The most common process of estimating the best model parameters, maximum likelihood estimation, is covered in the next section.

The kriging model is a spatial Gaussian process model. It defines the probability distribution of the model output over the model's domain with a Gaussian or normal distribution defined by the expected value given by Eq. (13) and variance given by Eq. (5). The mean square error (MSE), or variance of the estimate $\hat{y}(\mathbf{x})$ from Eq. (5) can be restated as

$$\mathrm{MSE}[\hat{y}(\mathbf{x})] = \sigma^2\left(1 - 2\boldsymbol{\lambda}^{\mathrm{T}}(\mathbf{x})\mathbf{r}(\mathbf{x}) + \boldsymbol{\lambda}^{\mathrm{T}}(\mathbf{x})\mathbf{R}\boldsymbol{\lambda}(\mathbf{x})\right), \tag{15}$$

where σ^2 is termed the *process variance* and is defined as the variance of the residuals. By substituting the estimate of $\boldsymbol{\lambda}(\mathbf{x})$ from Eq. (12) into Eq. (15), the following equation for the variance of the estimate results

$$\mathrm{MSE}[\hat{y}(\mathbf{x})] = \sigma^2\left(1 - \begin{bmatrix}\mathbf{f}^{\mathrm{T}}(\mathbf{x}) & \mathbf{r}^{\mathrm{T}}(\mathbf{x})\end{bmatrix}\begin{bmatrix}\mathbf{0} & \mathbf{F}^{\mathrm{T}} \\ \mathbf{F} & \mathbf{R}\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{f}(\mathbf{x}) \\ \mathbf{r}(\mathbf{x})\end{bmatrix}\right). \tag{16}$$

The estimates at the observations are returned exactly because the kriging model still interpolates the observations; the MSE at these points is zero since there is no uncertainty in the observations of a deterministic computer model. As an unobserved point, x, moves away from the observations, the second component of Eq (13) approaches zero, yielding the generalized least squares estimate, and the uncertainty in the estimate approaches its maximum value, namely, the process variance σ^2.

2.2 Maximum Likelihood Estimation

Most application of the Design and Analysis of Computer Experiments (DACE) [15] use the statistics-based method of Maximum Likelihood Estimation (MLE) as an objective estimator of the regression function coefficients, process variance, and spatial correlation function (SCF) parameters, $\boldsymbol{\gamma} = \left\{\boldsymbol{\beta}, \sigma^2, \boldsymbol{\theta}\right\}$, that are most consistent with the observed data [14, 16, 17]. MLE assumes that the residuals have a known probability distribution shape, which in most cases is the Gaussian probability distribution. The likelihood of the model parameters, given both the set of observations and the model's form, is defined as the probability of the n observations $\mathbf{y}$ given the model parameters $\boldsymbol{\gamma}$.

$$L(\boldsymbol{\gamma}|\mathbf{y}) = p(\mathbf{y}|\boldsymbol{\gamma}) = \prod_{i=1}^{n} p(y_i|\boldsymbol{\gamma}) \tag{17}$$

If the output distribution of the computer model comes from a Gaussian distribution, then the likelihood of the model parameters, $\boldsymbol{\gamma}$, is defined as a multivariate normal distribution of the n observations of $\mathbf{y}$ given the model parameters, $\boldsymbol{\gamma}$, and is given as

$$L(\boldsymbol{\gamma}|\mathbf{y}) = p(\mathbf{y}|\boldsymbol{\gamma}) = \frac{1}{\sqrt{(2\pi\sigma^2)^n |\mathbf{R}_{\boldsymbol{\theta}}|}} e^{-\frac{(\mathbf{y}-\mathbf{F}\boldsymbol{\beta})^{\mathrm{T}}\mathbf{R}_{\boldsymbol{\theta}}^{-1}(\mathbf{y}-\mathbf{F}\boldsymbol{\beta})}{2\sigma^2}}. \tag{18}$$

where the term $\mathbf{R}_{\boldsymbol{\theta}}$ includes the subscript θ to emphasize the fact that the correlation matrix is a function of the correlation range parameter.

The goal of the MLE method is to maximize the probability of all of the observations, given the model parameters, $\boldsymbol{\gamma}$. The multivariate normal likelihood function can be difficult to maximize due to potentially large flat regions of near zero values. To improve the optimization process, the logarithm of the likelihood function is taken, since the maximum of the logarithm of the likelihood occurs at the same location as the maximum of the likelihood. The logarithm of the multivariate Gaussian likelihood function is

$$\ell(\boldsymbol{\gamma}|\mathbf{y}) = -\frac{n}{2}\ln\left[2\pi\sigma^2\right] - \frac{1}{2}\ln\left[|\mathbf{R}_{\boldsymbol{\theta}}|\right] - \frac{1}{2\sigma^2}(\mathbf{y}-\mathbf{F}\boldsymbol{\beta})^{\mathrm{T}}\mathbf{R}_{\boldsymbol{\theta}}^{-1}(\mathbf{y}-\mathbf{F}\boldsymbol{\beta}). \tag{19}$$

By taking the derivative of the log-likelihood equation (Eq. (19)) with respect to $\boldsymbol{\beta}$ and σ^2 and solving for zero, the closed-form solution for the optimal value of $\boldsymbol{\beta}$ is

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{F}^{\mathrm{T}}\mathbf{R}_{\boldsymbol{\theta}}^{-1}\mathbf{F}\right)^{-1}\mathbf{F}^{\mathrm{T}}\mathbf{R}_{\boldsymbol{\theta}}^{-1}\mathbf{y}, \tag{20}$$

which matches the result of Eq. (14) found by solving for the least square error. The MLE solution for the process variance σ^2 is

$$\hat{\sigma}^2 = \frac{1}{n}\left(\mathbf{y}-\mathbf{F}\hat{\boldsymbol{\beta}}\right)^{\mathrm{T}}\mathbf{R}_{\boldsymbol{\theta}}^{-1}\left(\mathbf{y}-\mathbf{F}\hat{\boldsymbol{\beta}}\right). \tag{21}$$

A closed-form solution does not exist for the optimal parameters of most common SCFs, thus requiring numerical optimization. In order to reduce the number of model parameters determined with the numerical optimization, the known optimal values of $\hat{\boldsymbol{\beta}}$ and $\hat{\sigma}^2$ from Eqs. (20) and (21) are substituted into Eq. (19), resulting in the profile log-likelihood equation (Eq. (22)) [18]. The profile log-likelihood is then optimized for the unknown correlation parameters, $\boldsymbol{\theta}$.

$$\begin{aligned}&\text{maximize } -\frac{n}{2}\ln\left[2\pi\hat{\sigma}^2\right] - \frac{1}{2}\ln\left[|\mathbf{R}_{\boldsymbol{\theta}}|\right] - \frac{n}{2} \\ &\text{subject to } \theta > 0\end{aligned} \tag{22}$$

Even with the reduction in the dimension of the optimization problem that results from using the profile log-likelihood function, the optimization process can still be a computationally expensive process. The computational expense can be attributed primarily to two calculations: the evaluation of the exp function n^2 times in the correlation matrix and the inversion of the correlation matrix. In addition to the potential computational

expense associated with each iteration of an optimization algorithm, the optimization of the log-likelihood equation can have several numerical difficulties. The three most prevalent issues are: (1) ill-conditioned correlation matrices, (2) multiple local optimum, and (3) long ridges of near optimal values [19]. The remainder of this paper presents optimization developments to alleviate some of the computational expense and to deal with the numerical difficulties

2.3 Optimization Algorithms

This section first provides a brief background on the gradient-based optimization algorithms that are commonly used with MLE parameter optimization. These algorithms include first and second order methods and combinations between the two. The section closes with a description of the three different variations to second order or Newton-Raphson based methods that are compared for their effectiveness at optimizing the log-likelihood equation.

There are two primary questions that need to be answered at each iteration of the algorithm: (1) Which direction to head? and (2) How far to go in that direction? The most logical answer to the first question is to head in the direction of the steepest ascent. By definition this is the gradient of the log-likelihood equation at the current values of the model parameters. The step size is often taken as the largest step that still results in an increase in the log-likelihood equation. There are many optimization algorithms that determine optimal step sizes, most of them requiring a line search. These methods can have difficulty converging rapidly near the optimum because of their tendency to zig-zag. If the maximum increase is found in the current direction, the next direction will be normal to the current direction. The conjugate gradient methods include information about previous gradients with the current gradient, smoothing out the zig-zag behavior and showing improved convergence performance.

A second order approximation can be used to improve upon some of the deficiencies of the linear methods. One of the most common methods used is the Newton-Raphson method. In the Newton-Raphson method, a quadratic surface

$$f(\boldsymbol{\theta}) = \mathbf{a} + \mathbf{b}^T\boldsymbol{\theta} - \tfrac{1}{2}\boldsymbol{\theta}^T\mathbf{C}\boldsymbol{\theta} \tag{23}$$

is fit to the log-likelihood function at the current model parameter values. The maximum value of this quadratic surface is given analytically as $\boldsymbol{\theta} = \mathbf{C}^{-1}\mathbf{b}$ where $-\mathbf{C}$ is the Hessian of the log-likelihood function and the gradient is equal to $\mathbf{b} - \mathbf{C}\boldsymbol{\theta}$. This solution provides both the direction and distance to travel along the direction. If the log-likelihood function were quadratic, this method would solve the optimization problem in a single step from any starting values.

In general, the log-likelihood function to be maximized is not quadratic, and as a result an iterative technique is needed to solve the maximization problem. Given the current values of the model parameters $\boldsymbol{\theta}^{(k)}$ and the approximated quadratic function at the current values, the next step $k+1$ is defined as

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - \left(\mathbf{H}^{(k)}\right)^{-1}\nabla f^{(k)} \tag{24}$$

where $\mathbf{H}^{(k)}$ and $\nabla f^{(k)}$ refer to the Hessian and the gradient evaluated at the current model parameter values $\boldsymbol{\theta}^{(k)}$. The primary drawback to this method is the need to evaluate both the gradient and the Hessian of the log-likelihood function at each iteration of the algorithm, with the Hessian generally being the most computationally expensive to evaluate. Quasi-Newton optimization methods look to approximate the Hessian by combining the gradients from previous iterations. One of the most popular methods is the BFGS (Broyden, Fletcher, Goldfarb, and Shannon) update formula.

In general, a steepest ascent algorithm will converge to the optimal value more rapidly when the starting value is far from the optimal value than a Newton-Raphson method. The Newton-Raphson method will converge faster as the current value of the iteration closes in on the optimal value [20]. The Levenberg-Marquardt method [2] provides a heuristic to transition from the steepest ascent method to the Newton-Raphson method. The iteration of Eq. 24 can be altered as

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - \left(\mathbf{H}^{(k)} + \tau^{(k)}\mathbf{I}\right)^{-1}\nabla f^{(k)} \tag{25}$$

where $\tau^{(k)}$ as a scalar that indirectly determines the step size and $\mathbf{I}$ is the identity matrix. For small values of $\tau^{(k)}$, the method approximates the Newton-Raphson method, and for large values, a small step is taken in nearly the direction of steepest descent(ascent). In general, the algorithm proceeds by taking progressively smaller values of $\tau^{(k)}$ for each iteration that results in an improved value and increases $\tau^{(k)}$ if it results in a worse value. The Levenberg-Marquardt method is used for all three methods for calculating the Hessian that are compared in this work.

The first variation explored in this work is to use the full Hessian of the log-likelihood equation in the Levenburg-Marquardt method. The inverse of the Hessian of the log-likelihood equation (Eq. 19), evaluated at the MLE selected model parameter values, is defined in statistics as the observed Fisher information matrix. Each observation of the original computer model provides information about the process embodied by the computer model. The amount of information available to select the correct model parameters can be quantified by the observed Fisher information matrix [21].

The second variation explored in this work is to use the expected value of the information matrix as an approximation to the Hessian of the log-likelihood equation. The expected value of the information matrix is defined as the expected value of the Hessian matrix. The inverse of the expected information matrix provides the asymptotic estimate of the variance of the model parameters. The evaluation of the expected value of the Hessian can be simplified by the relationship (for a proof see the appendix of Searle et. al [22])

$$E\left(\frac{\partial^2 \ell}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T}\right) = -E\left(\frac{\partial \ell}{\partial\boldsymbol{\theta}}\frac{\partial \ell}{\partial\boldsymbol{\theta}^T}\right). \tag{26}$$

The result of this relationship is that the Hessian can be estimated by taking the expected value of the outer product of the gradient. The use of this approximation for the Hessian in the Newton-Raphson optimization method is termed the Scoring

method [23]. Further details on the expected value of the Hessian matrix are given in the next section.

A third variation explored in this work is to use an approximation to the Hessian that is determined by calculating only the diagonal elements of the expected Hessian matrix and assigning zero to all off-diagonal values. This approximation greatly simplifies the calculation of the inverse to the Hessian as its inverse is the matrix that results from inverting each diagonal element, an order d operation rather than an order d^3 operation, where d is the number of covariance parameters.

3 DERIVATION OF GRADIENT AND HESSIAN

The gradient is a vector that is defined as the first partial derivative of the log-likelihood function. The Hessian is a matrix that is defined as the second partial derivative. This section provides the derivation of the analytical first and second partial derivatives of the log-likelihood equation when the Gaussian function is used as the spatial correlation function.

A finite difference approximation is typically used to calculate the gradient and Hessian of the log-likelihood function. At a minimum, the gradient requires $d+1$ log-likelihood function evaluations with different correlation parameters. The calculation of the Hessian requires d^2+1 log-likelihood function evaluations. There are two computationally expensive calculations needed to evaluate the log-likelihood function. Each function evaluation requires the calculation of the n^2 spatial correlation function at each combination of the n observations and the inversion of the correlation matrix, $\mathbf{R}$.

The remainder of this section provides a closed-form analytical solution for the gradient and Hessian that do not require additional evaluations of the spatial correlation function and the inversion of the correlation matrix other than at the current values of the correlation parameters. These forms are exact rather than the finite difference approximations used in the past and are much more computationally efficient to evaluate since they do not require the calculation and inversion of many different correlation matrices $\mathbf{R}$.

3.1 First Partial Derivatives

If the model parameters $\boldsymbol{\gamma}$ are divided into the coefficients of the regressors $\boldsymbol{\beta}$ and the parameters $\boldsymbol{\theta}$ that define the covariance matrix (Eq. 11), then the vector of the first partial derivatives of the log-likelihood function ℓ, with respect to $\boldsymbol{\gamma}$, can be written as $\ell^{(1)} = \{\ell_\beta^{\mathrm{T}}, \ell_\theta^{\mathrm{T}}\}^{\mathrm{T}}$, which is also known as the *score* function. The partial derivative of the log-likelihood function with respect to the regressor coefficients is

$$\ell_\beta = -\mathbf{F}^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{F}\boldsymbol{\beta} + \mathbf{F}^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{y} \tag{27}$$

and the ith element of ℓ_θ is

$$(\ell_\theta)_i = -\tfrac{1}{2}\mathrm{tr}\left(\mathbf{V}^{-1}\mathbf{V}_{\mathrm{i}}\right) - \tfrac{1}{2}\mathbf{W}^{-1}\mathbf{V}^{\mathrm{i}}\mathbf{W} \tag{28}$$

where the inverse of the covariance matrix is $\mathbf{V}^{-1} = \sigma^{-2}\mathbf{R}^{-1}$, $\mathbf{V}_i = \partial\mathbf{V}/\partial\theta_i$ is a matrix that represents the partial derivative of the covariance matrix with respect to the covariance parameter θ_i, $\mathbf{V}^i = \partial\mathbf{V}^{-1}/\partial\theta_i = -\mathbf{V}^{-1}\mathbf{V}_i\mathbf{V}^{-1}$ is the inverse of the partial derivative of the covariance matrix for each covariance parameter, and $\mathbf{W} = \mathbf{y} - \mathbf{F}\boldsymbol{\beta}$ is the vector of residuals.

The partial derivative of the covariance matrix $\mathbf{V}$ with respect to the process variance σ^2 is the correlation matrix $\mathbf{R}$.

$$\frac{\partial\mathbf{V}}{\partial\sigma^2} = \mathbf{R} \tag{29}$$

The partial derivatives of the covariance matrix $\mathbf{V}$ with respect to the correlation parameters are determined by evaluating the partial derivative of the covariance function at each of the k observations in $\mathbf{X}$. The partial derivative of the kth row (or column) of the covariance function is

$$\frac{\partial\mathbf{V}_k}{\partial\theta_i} = \frac{\partial\mathbf{v}(\mathbf{X}_k)}{\partial\theta_i}. \tag{30}$$

The resulting partial derivative matrix with respect to each correlation parameters is positive semi-definite and symmetric about the diagonal, just like the correlation matrix $\mathbf{R}$. Unlike the correlation matrix, the diagonal of the partial derivative of the covariance matrix consists of zeros since, by inspection of Eq (32), the derivative of the variance of an observations is zero.

The partial derivative of the covariance function with respect to the process variance σ^2 is simply the correlation function,

$$\frac{\partial\mathbf{v}(\mathbf{x})}{\partial\sigma^2} = \mathbf{r}(\mathbf{x}). \tag{31}$$

The $i = 1,\ldots,d$ partial derivatives of the covariance function with respect to the correlation parameters are defined as

$$\frac{\partial\mathbf{v}(\mathbf{x})}{\partial\theta_i} = \frac{2\sigma^2\mathbf{r}(\mathbf{x})\odot(\mathbf{X}_i - \mathbf{x}_i)^2}{\theta_i^3} \tag{32}$$

where $\mathbf{X}_i$ is the ith column of the matrix of inputs $\mathbf{X}$, $\mathbf{x}_i$ is the ith dimension of an unobserved point in the domain Ω, and $\odot$ is an operator indicating the element-wise multiplication of two vectors. The resulting partial derivatives of the covariance parameters are n-element vectors corresponding to each of the observations.

3.2 Second Partial Derivatives

The second derivative matrix or Hessian of the log-likelihood function can be partitioned into four sub-matrices [16] as

$$\ell^{(2)} = \begin{bmatrix} \ell_{\beta\beta} & \ell_{\beta\theta} \\ \ell_{\beta\theta}^{\mathrm{T}} & \ell_{\theta\theta} \end{bmatrix}, \tag{33}$$

where $\ell_{\beta\beta} = \mathbf{F}^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{F}$, $\ell_{\beta\theta}$ has ith column $-\mathbf{F}^{\mathrm{T}}\mathbf{V}^{i}\mathbf{F}\boldsymbol{\beta} + \mathbf{F}^{\mathrm{T}}\mathbf{V}^{i}\mathbf{y}$, and $\ell_{\theta\theta}$ has (i, j)th term

$$-\frac{1}{2}\left(\mathrm{tr}\left(\mathbf{V}^{-1}\mathbf{V}_{\mathrm{ij}} + \mathbf{V}^{\mathrm{i}}\mathbf{V}_{\mathrm{j}}\right) + \mathbf{W}^{\mathrm{T}}\mathbf{V}^{\mathrm{ij}}\mathbf{W}\right) \tag{34}$$

where $\mathbf{V}_{ij} = \partial^2\mathbf{V}\big/\partial\theta_i\partial\theta_j$, and

$$\mathbf{V}^{ij} = \frac{\partial^2\mathbf{V}^{-1}}{\partial\theta_i\partial\theta_j} = \mathbf{V}^{-1}\Big(\mathbf{V}_i\mathbf{V}^{-1}\mathbf{V}_j + \mathbf{V}_j\mathbf{V}^{-1}\mathbf{V}_i - \mathbf{V}_{ij}\Big)\mathbf{V}^{-1}. \quad (35)$$

There are four cases that must be considered when determining the second partial derivatives of the covariance function $\mathbf{V}_{ij}$ analytically. The first case is $\partial\sigma^2\partial\sigma^2$. This case is trivial since the first partial derivative (see Eq. (31)) is not a function of the process variance σ^2. Therefore, the following results

$$\frac{\partial^2\mathbf{v}(\mathbf{x})}{\partial\sigma^2\partial\sigma^2} = \mathbf{0} \quad (36)$$

where, in this equation, $\mathbf{0}$ is a vector of n zeroes.

As a result of Eq. (36), the second derivative of the covariance matrix with respect to $\partial\sigma^2\partial\sigma^2$ is an $(n\times n)$ matrix of zeroes.

$$\frac{\partial^2\mathbf{V}}{\partial\sigma^2\partial\sigma^2} = \mathbf{0} \quad (37)$$

The second case of interest is $\partial\sigma^2\partial\theta_i$. Since differentiation is a linear operation, the following equivalency of expressions of the derivative can be made.

$$\frac{\partial^2\mathbf{v}(\mathbf{x})}{\partial\sigma^2\partial\theta_i} = \frac{\partial^2\mathbf{v}(\mathbf{x})}{\partial\theta_i\partial\sigma^2} = \frac{\partial\mathbf{v}(\mathbf{x})}{\partial\theta_i}\frac{1}{\sigma^2} \quad (38)$$

which results from differentiating the results of Eq. (32) with respect to the process variance σ^2. The corresponding second partial derivative of the kth row (or column) of the covariance matrix is given as

$$\frac{\partial\mathbf{V}_k}{\partial\sigma^2\partial\theta_i} = \frac{\partial\mathbf{V}_k}{\partial\theta_i\partial\sigma^2} = \frac{\partial\mathbf{v}(\mathbf{X}_k)}{\partial\sigma^2\partial\theta_i}. \quad (39)$$

The third case of interest is $\partial\theta_i\partial\theta_j$, where $i \neq j$. The partial derivative of Eq. (32) with respect to a different correlation range parameter is given as

$$\frac{\partial^2\mathbf{v}(\mathbf{x})}{\partial\theta_i\partial\theta_j} = \frac{\partial\mathbf{v}(\mathbf{x})}{\partial\theta_i}\odot\frac{2\big(\mathbf{X}_j-\mathbf{x}_j\big)^2}{\theta_j^3} \quad (40)$$

Following the same pattern as the previous two cases, the partial derivative of the kth row (or column) of the covariance matrix is given as

$$\frac{\partial\mathbf{V}_k}{\partial\theta_i\partial\theta_j} = \frac{\partial\mathbf{v}(\mathbf{X}_k)}{\partial\theta_i\partial\theta_j}. \quad (41)$$

The last case of second partial derivatives of interest in this study is $\partial\theta_i\partial\theta_i$. The partial derivative of Eq. (32) with respect to the same correlation parameter is given as

$$\frac{\partial^2\mathbf{v}(\mathbf{x})}{\partial\theta_i\partial\theta_i} = \frac{\partial\mathbf{v}(\mathbf{x})}{\partial\theta_i}\odot\left(\frac{2(\mathbf{X}_i-\mathbf{x}_i)^2}{\theta_i^3} - \frac{\mathbf{3}}{\theta_i}\right) \quad (42)$$

where $\mathbf{3}$ is a vector of n threes. The second partial derivative of the kth row (or column) of the covariance matrix is given as

$$\frac{\partial^2\mathbf{V}_k}{\partial\theta_i\partial\theta_i} = \frac{\partial^2\mathbf{v}(\mathbf{X}_k)}{\partial\theta_i\partial\theta_i}. \quad (43)$$

3.3 Expected Value of Hessian

The expected value of the Hessian can be determined by using $E(\mathbf{y}) = \mathbf{F}\boldsymbol{\beta}$ and hence $E(\mathbf{y}-\mathbf{F}\boldsymbol{\beta}) = E(\mathbf{W}) = \mathbf{0}$ and the relationship from Eq. 26. The resulting expected value of the Hessian

$$E\begin{bmatrix}\ell_{\beta\beta} & \ell_{\beta\theta}\\ \ell_{\beta\theta}^{\mathrm{T}} & \ell_{\theta\theta}\end{bmatrix} = -\begin{bmatrix}\mathbf{F}^T\mathbf{V}^{-1}\mathbf{F} & \mathbf{0}\\ \mathbf{0} & \frac{1}{2}\mathrm{tr}\Big(\mathbf{V}^{-1}\mathbf{V}_i\mathbf{V}^{-1}\mathbf{V}_j\Big)_{i,j=0}^{d}\end{bmatrix} \quad (44)$$

depends only on the covariance matrix and its first partial derivatives with respect to the covariance parameters. The expected value of the Hessian matrix is much more computationally efficient to calculate than the full Hessian. When the kriging model is able to correctly specify the distribution of the observations, i.e. the residuals are well represented with a zero mean Gaussian shape, the full Hessian evaluated at the MLE parameter values and the expected Hessian matrices should be nearly identical [24]. The degree to which the residuals are well represented with a Gaussian distribution can be estimated by $\ell_{\beta\theta}$ term of Eq. 33 being nearly equal to $\mathbf{0}$.

3.4 Implementation

The gradient and Hessian of the log-likelihood function are relatively simply to represent analytically for the Gaussian correlation function. The actual evaluation of these derivatives can be difficult to encode on a computer since they require the use of both 3-D and 4-D tensors for the first and second partial derivatives. This section provides a few recommendations to reduce the computational burden and simplify the implementation of the algorithms.

The first recommendation is calculate and store the 3-D tensor $(n\times n\times d)$ *ipd* that represents the interpoint distances in each dimension. All of these distances are greater than or equal to zero. The *ipd* tensor needs to be calculated only once; it depends only upon the observations and not the kriging model parameters. This tensor of interpoint distances is also used in the calculation of the first partial derivative of the covariance matrix with respect to the correlation parameters (Eq. 30) and for the second partial derivatives (Eqs. 40 and 42).

The second recommendation is that the correlation matrix $\mathbf{R}$ and the inverse to the correlation matrix $\mathbf{R}^{-1}$ need to be calculated only once per current value of the correlation parameters. The correlation matrix is a function of the *ipd* tensor and the correlation parameters $\boldsymbol{\theta}$. The calculation of the gradient and the Hessian does not require the inversion of any other matrices. This is a major improvement of computational efficiency of all of the optimization methods compared in this work. This improvement is more prominent as more dimensions are used in the model.

The last recommendation to reduce the computational burden involved with calculating the Hessian is to first calculate and store the partial derivative of the covariance matrix $\mathbf{V}$ with respect to each of the covariance parameters. This matrix is used extensively in the calculation of most elements of the Hessian matrix.

4 DETAILS REQUIRED FOR OPTIMIZATION

This section provides additional details needed to implement the optimization algorithms as identified in the previous section. The first detail is to scale the input domain to the unit hypercube. In most cases a linear mapping is sufficient. In situations where the spatial covariance is heteroscedastic, a nonlinear mapping may be preferred. The second detail is the implementation of a constrained optimization using unconstrained optimization methods. The third detail describes the implementation of the Levenburg-Marquardt heuristic used in this work. The last detail defines the method used to sample the range of the correlation parameters in a more efficient manner during a multi-start optimization.

The Newton-Raphson method is an unconstrained optimization algorithm. In this implementation, the values of the correlation range parameters, as defined in the spatial correlation function in Eq. 3, are constrained to be within a minimum and maximum value. In this work they are taken to be half the smallest distance between the observations and ten times the largest distance (in most cases with a unit hypercube this will be one) respectively. An additional constraint is also added such that the resulting correlation matrix must be well-conditioned in order to calculate its inverse accurately.

An unconstrained optimization method can be used to perform constrained optimization by using a penalty function to "penalize" the cost function for exceeding the optimization problem's constraints. A penalty function method does not perform properly while optimizing the log-likelihood function. It frequently "runs away" or becomes unstable. Evaluating the Gaussian spatial correlation function for negative correlation range values is symmetric with positive correlation range values, leaving two potential solutions when only the positive one is correct. Additionally, allowing correlation parameters that result in an ill-conditioned correlation matrix results in numerical in accuracies while inverting the correlation matrix. The results of these numerical issues is the correlation parameters must always be kept within the optimization problems constraints.

The minimum and maximum constraints are handled by calculating the next model parameters $\boldsymbol{\theta}_{k+1}$ and testing their values. If the current value is not on a constraint and the next one exceeds its minimum or maximum constraints, then the distance of the step in all dimensions is scaled to locate the next point on the constraint. If the current value is on a constraint and the next step exceeds its minimum or maximum constraints, then the distance on the constrained dimension is set to zero to remain on the constraint and all values in the other dimensions remain unchanged. The result of this modification to the step is the optimization algorithm will travel along the minimum or maximum constraint. If the algorithm stops moving, i.e. a resulting step length of very near zero, then the iterations stop and the resulting value is returned. The optimal value will often be found on a constraint.

The ill-conditioned correlation matrix constraint is tested after the minimum and maximum correlation parameter constraints are tested. The calculation of the correlation matrix may fail if the correlation parameters are not within their minimum and maximum constraints. The logarithm of the condition number (the ratio of the largest to smallest eigenvalues) should be less than the machine precision. This constraint was set at 15 for this work.

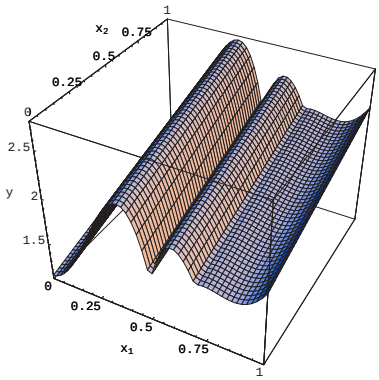
The scalar Levenberg-Marquardt parameter $\tau^{(k)}$ is initially set to 0.001 given the input domain has been scaled to the unit hypercube. For every iteration of the optimization that results in a better result, $\tau^{(k)}$ is decreased by a factor of ten, if it is worse then it is increased by a factor of ten. The scalar parameter $\tau^{(k)}$ is added to each of the diagonal element of the Hessian matrix as identified in Eq. 25. The Hessian modified in Eq. 25 must be inverted to determine the next value of the model parameters. In many situations where the log-likelihood function is multi-modal or has very flat regions of nearly identical values, the Hessian may become ill-conditioned. The logarithm of its condition number is also tested to be less than eight in this case. If the Hessian is ill-conditioned, then the Levenberg-Marquardt scalar parameter is temporarily set to one, providing a modified Hessian that can be inverted and results in a small step in the direction of gradient.

The iterating of the Newton-Raphson optimization algorithm is stopped when either a convergence test is passed or a limit in the number of iterations is exceeded. The convergence test is defined as the mean square of the log-likelihood function gradient being within a convergence limit of 0.001 four consecutive times. The maximum number of iterations is set to be 100. The algorithm will also stop iterating if the step size of the current iteration is less than 1×10^{-8} four times in a row.

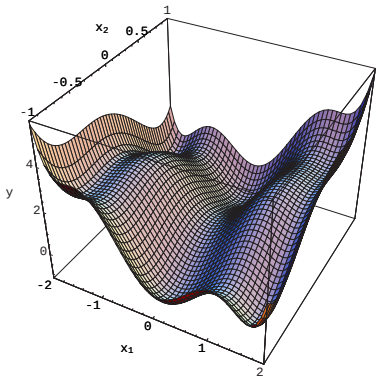
All of the versions of the Newton-Raphson optimization algorithm compared in this work are unable to determine a global optimum; they can only determine an optimum that is local to the starting point of the algorithm. As a result, a multi-start strategy is used to improve the probability that the global maximum has been found. The correlation range parameters are constrained to be within the minimum and maximum constraints as defined previously. In general, it is inefficient to uniformly sample this space randomly; an initial point will frequently be chosen that results in an ill-conditioned correlation matrix. The log-likelihood function will typically have more local optimum with smaller correlation range parameters. As a result, starting values are randomly generated on a logarithmic scale. Given a uniform [0,1] random sample Z and a minimum correlation range of 0.01, the following equation is used to generate the starting values for each correlation parameter θ

$$\theta = 10^{Z \log_{10}(\text{maxCorr}*100)-2} \tag{45}$$

where maxCorr refers to the maximum correlation constraint. If the starting values result in a correlation matrix that is ill-conditioned, they are thrown out and a new random sample is taken. Each of the gradient-based optimization algorithms are restarted with new samples until either: (1) the same maximum value is returned three more times, or (2) the restart limit of 100 samples is exceeded.



(a) Test Function 1



(b) Six-Hump Camelback Function

Figure 1: Plots of the 2-D Test Functions

5 RESULTS

The use of the analytical gradient and Hessian for the kriging model are compared for their computational expense. The three variations of the Newton-Raphson optimization methods (see Section 2.3) are compared based upon the results of optimizing two different sets of data. The test examples include 2 two- and a five-dimensional data set. The comparisons include the average number of iterations and required to converge to a local optimum, the average number of restarts required to converge to a global optimum, and the total average iterations required to converge to a global optimum. The comparisons will also include an estimate of the computational expense associated with estimating the Hessian used in the three variations.

5.1 Example Problems

The first two-dimensional test function was introduced by Osio and Amon [25]. This function is linear in one dimension and highly nonlinear in the second dimension. It is approximated using 21 observations taken from an optimized Latin hypercube design [8]. The second two-dimensional test function is the Six-Hump Camelback function [26]. This function is multi-model and is difficult to fit with a traditional regression function. A representation of both two-dimensional examples can be seen in Figure 1. The five-dimensional test function is the analysis results of a gas generation system [27]. It has five inputs that describe the geometry of the system and its operating conditions. It returns the volume of gas generated by the system. For this function, a 40-point Latin hypercube was used to generate observations.

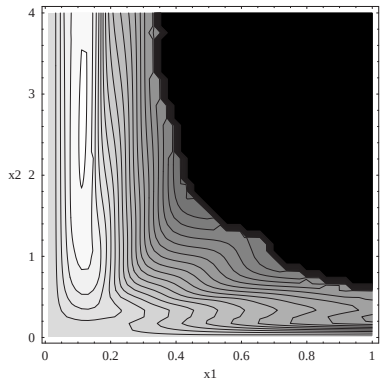
A non-trivial trend function is used in all of the examples. For the first 2-D example, the best trend function was chosen by using the corrected Akaike's Information Criterion (cAIC) [9]. The trend function includes a constant term and a linear term for the x2 dimension. For the second 2-D example, the best trend function includes the constant term and linear and quadratic terms for x1. It lacks any terms that include x2. For the 5-D example, a good trend function was chosen, but not the best based on the cAIC. One was chosen that resulted in optimal correlation ranges that do not fall on constraints. The trend function consists of a constant term, and the linear and quadratic terms for x2, x3, and x4. There are no cross terms.

Contour plots of the log-likelihood function for the 2-D example problems are shown in Figure 2. The larger likelihood values are lighter colored. The black region in the plot indicates the region of correlation values that result in an ill-conditioned correlation matrix. The two plots only represents a portion of the total range $[.01, 10]^2$ that is searched for the optimal value of the correlation parameters. Both plots are characterized by having three valleys that lead to the origin. The valleys for the six-hump camelback example are more pronounced. The result of having these valleys is a gradient-based optimization will not be able to jump out of them until they approach the origin. By observing the plot for test function 1, it can be seen that there is a long ridge of near maximum values along x1=0.11. It can also be seen that for small values of x2, the gradient will push the optimization algorithm to the minimum constraint for x2 and then slowly travel along it to either larger or smaller values of x1. If it pushes to smaller value of x1, then the optimization may eventually converge to the global maximum. Otherwise, the algorithm will tend to converge to a non-global solution or get stuck along a constraint boundary. The plot for the six-hump camelback example has some similarities to the other 2-D example in that it also has a long ridge along x1=0.09. It is impossible to provide a similar representation of the log-likelihood function as shown in Figure 2 for the 5-D example problem. The resulting log-likelihood function does have many local maxima: both internal to the range of model parameters and along the constraints of the correlation parameters.

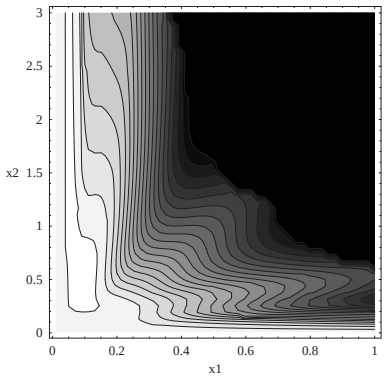
5.2 Comparison of Optimization Methods

The three methods of calculating or estimating the Hessian as part of a Newton-Raphson optimization of the log-likelihood equation is compared in this section. The methods are compared based upon: 1) the average number of iterations required to converge to an answer, 2) the average number of starts required to converge to the optimal answer 4 times, 3) the average number of iterations to determine the optimal answer, 4) the average CPU time per iteration, and 5) the average total CPU time to determine the optimal answer. The results were generated by executing the algorithms enough times to record more than 500 starts of the algorithms and are shown in Tables 1- 3. The experiments were run on a 1.3MHz Pentium M with 1 GB of RAM.

The results of this comparison of three examples give relatively consistent results: the Simplified Scoring method appears to perform the best. In all of the cases on average it converges to an optimal solution in the fewest number of iterations (the first row in the tables). The Expected Hessian case tends to converge to the correct optimum with the fewest starts, but it takes more



(a) Test Function 1



(b) Six-Hump Camelback Function

Figure 2: Contour Plots of the Log-likelihood of 2-D Examples

Table 1: Comparison of Optimization Method Options for 2-D Example 1

Metric	Full Hess.	Exp. Hess.	Sim. Scoring
iters. to converge	28.4	41.7	17.0
starts to optimal	10.70	6.59	7.89
total iters. to opt.	305.2	278.2	135.1
time (ms) per iter.	12.10	5.40	4.58
time (ms) to opt.	3690	1500	618

iterations to converge to an answer in both 2-D examples. In the 5-D example, the Expected Hessian method converged faster than the Simplified Scoring method. The minimum number of samples possible in each start is 4 and the maximum limit of 101. The maximum limit may tend to bias the results slightly to be lower than they should be. Without using a stopping criterion for the maximum limit of the iterations of 101, it was possible for the Newton-Rahpson algorithms to continue without converging in some instances. The Full Hessian for the 5-D example never converged to the same answer within the maximum limit of 101 iterations.

One possible explanation for the observed difference between the Full Hessian and Expected Hessian results lies in the Expected Hessian being a negative semi-definite matrix over the entire range of correlation parameters. The actual Hessian may not be negative semi-definite at all locations as can be seen in the contour plots of Figure 2. This can cause instability in the Newton-Raphson algorithm. As a result, even though the parameter values are constrained to be within valid ranges, the resulting parameter values may still be located on a constraint that has a very little gradient in other directions, causing the algorithm to stall. The positive results of going to a constraint very quickly is the algorithm will stop after typically 4 or 5 iterations and move on quickly to another starting value.

Table 2: Comparison of Optimization Method Options for 2-D Example 2

Metric	Full Hess.	Exp. Hess.	Sim. Scoring
iters. to converge	30.3	24.0	15.2
starts to optimal	9.88	6.96	7.17
total iters. to opt.	300.6	168.3	108.8
time (ms) per iter.	11.81	6.73	5.09
time (ms) to opt.	3550	1130	554

Table 3: Comparison of Optimization Method Options for 5-D Example

Metric	Full Hess.	Exp. Hess.	Sim. Scoring
iters. to converge	63.8	59.3	53.7
starts to optimal	n/a	19.30	44.5
total iters. to opt.	n/a	1140	2390
time (ms) per iter.	194	74.7	27.3
time (s) to opt.	n/a	85.2	65.3

The results of the difference between the Expected Hessian and the Simplified Scoring methods may be explained by the amount of correlation that exists between the range parameters. If the range parameters are relatively independent of each other, then the Expected Hessian matrix will have near zero values on its off-diagonal terms. The result is the Expected Hessian matrix is well approximated by the Simplified Scoring method. This appears to be the case for both 2-D examples, but is not the case for the 5-D example.

The execution times tell a similar story as the number of iterations. The average time per iteration decreases from the Full Hessian to the Expected Hessian to the Simplified Scoring methods as expected. As more dimensions are included, this difference becomes more pronounced. For both 2-D examples, this result amplifies the result that the Simplified Scoring method appears to determine the global optimum the fastest of the three methods. For the 5-D example, the improved speed of execution per iteration resulted in the average time to determine the global optimum to be the lowest for the Simplified Scoring method even though it must perform more iterations in the process (2390 vs. 1140).

6 CONCLUSIONS

This paper presents an analytical form of the gradient and Hessian for the log-likelihood equation of a kriging model that uses a Gaussian spatial correlation function. These analytical forms are demonstrated for the computational efficiency and effectiveness on three example problems using three variations of

the Newton-Raphson optimization method. It was found that a Simplified Scoring method was the most computationally efficient method to determine the global optimum of the log-likelihood function.

These results do not leave the calculation of the Full Hessian without a purpose in the creation of kriging models. The Full Hessian can be used to estimate the covariance matrix of the model parameters, the observed Fisher information matrix. From this, the uncertainty (variance) in the estimate of the model parameters can be determined, their *p*-values calculated, and a *t*-test performed to determine if a model parameter should be included in the model. In general, this test would not be used to remove correlation range parameters, but it can be used to remove trend function parameters [9].

Future work should explore developing a heuristic to switch from the Simplified Scoring method to the Expected or Full Hessian as the Newton-Raphson method converges. Once the algorithm approaches the optimum, the use of the Full Hessian should in a better approximation of the local behavior of the log-likelihood function.

7 Acknowledgment

The author would like to thank Dr. Kam Ng of ONR 333 for his generous support on Contract No. N00014-00-G-0058.

REFERENCES

[1] Simpson, T. W., Maurey, T. M., Korte, J. J. and Mistree, F., 2001, "Kriging Metamodels for Global Approximation in Simulation-Based Multidisciplinary Design Optimization", *AIAA Journal,* **39**(12), pp. 2233–2241.

[2] Marquardt, D. W., 1963, "An Algorithm for Least-Squares Estimation of Nonlinear Parameters", *Journal of the Society for Industrial and Applied Mathematics,* **11**(2), pp. 431–441.

[3] Krige, D. G., 1951, "A Statistical Approach to Some Mine Valuations and Allied Problems at the Witwatersrand". *Master's Thesis*, University of the Witwatersrand.

[4] Matheron, G., 1963, "Principles of Geostatistics", *Economic Geology,* **58**, pp. 1246–1266.

[5] Goovaerts, P., 1997, *Geostatistics for Natural Resources Evaluation,* Applied Geostatistics Series, Oxford University Press, New York.

[6] Goldberger, A. S., 1962, "Best Linear Unbiased Prediction in the Generalized Linear Regression Model", *Journal of the American Statistical Association,* **57**(298), pp. 369–375.

[7] Welch, W. J., Buck, R. J., Sacks, J., Wynn, H. P., Mitchell, T. J. and Morris, M. D., 1992, "Screening, predicting, and computer experiments", *Technometrics,* **34**(1), pp. 15–25.

[8] Martin, J. D. and Simpson, T. W., 2005, "On the Use of Kriging Models to Approximate Deterministic Computer Models", *AIAA Journal,* **43**(4), pp. 853–863.

[9] Martin, J. D. and Simpson, T. W., 2006, "A Methodology to Manage Uncertainty During System-Level Conceptual Design", *ASME Journal of Mechanical Design,* **128**(4), pp. 959–968.

[10] Koehler, J. R. and Owen, A. B., 1996, "Computer Experiments", In *Handbook of Statistics*, S. Ghosh and C. R. Rao, eds., Elsevier Science, New York, pp. 261–308.

[11] Stein, M. L., 1999, *Interpolation of Spatial Data : Some Theory for Kriging*, Springer Series in Statistics, Springer-Verlag, New York.

[12] Simpson, T. W., Peplinski, J., Koch, P. N. and Allen, J. K., 2001, "Metamodels for Computer-Based Engineering Design: Survey and Recommendations", *Engineering with Computers,* **17**(2), pp. 129–150.

[13] Booker, A. J., Conn, A. R., Dennis Jr., J. E., Frank, P. D., Trosset, M. and Torczon, V., 1995, "Global Modeling for Optimization: Boeing/IBM/Rice Collaborative Project 1995 Final Report", *ISSTECH-95-032*, The Boeing Company, Bellevue, WA.

[14] Currin, C., Mitchell, T. J., Morris, M. D. and Ylvisaker, D., 1991, "Bayesian Prediction of Deterministic Functions, With Applications to the Design and Analysis of Computer Experiments", *Journal of the American Statistical Association*, **86**(416), pp. 953–963.

[15] Sacks, J., Welch, W. J., Mitchell, T. J. and Wynn, H. P., 1989, "Design and Analysis of Computer Experiments", *Statistical Science*, **4**(4), pp. 409–435.

[16] Mardia, K. and Marshall, R., 1984, "Maximum Likelihood Estimation of Models for Residual Covariance in Spatial Regression", *Biometrika*, **71**(1), pp. 135–146.

[17] Kitanidis, P. K., 1986, "Parameter Uncertainty in Estimation of Spatial Functions: Bayesian Analysis", *Water Resources Research,* **22**, pp. 499–507.

[18] Mardia, K. and Watkins, A. J., 1989, "On Multimodality of the Likelihood in the Spatial Linear Model", *Biometrika*, **76**(2), pp. 289–295.

[19] Warnes, J. J. and Ripley, B. D., 1987, "Problems with Likelihood Estimation of Covariance Function of Spatial Gaussian Processes", *Biometrika*, **74**(3), pp. 640–2.

[20] Hemmerle, W. J. and Hartley, H. O., 1973, "Computing Maximum Likelihood Estimates for the Mixed A. O. V. Model Using the W Transform", *Technometrics,* **15**(4), pp. 819–831.

[21] Fisher, R. A., 1925, "Theory of Statistical Estimation", *Proceedings of the Cambridge Philosophical Society*, **22**, pp. 700–725.

[22] Searle, S. R., C. G. and McCulloch, C. E., 1992, *Variance Components*, John Wiley & Sons, Inc., New York.

[23] Jennrich, R. I. and F., S. P., 1976, "Newton-Raphson and Realted Algorithms for Maximum Likelihood Variance Component Estimation", *Technometrics,* **18**(1), pp. 11–17.

[24] White, H., 1982, "Maximum Likelihood Estimation of Misspecified Models", *Econometrica,* **50**(1), pp. 1–26.

[25] Osio, I. G. and Amon, C. H., 1996, "An Engineering Design Methodology with Multistage Bayesian Surrogate and Optimal Sampling", *Research in Engineering Design,* **8**(4), pp. 189–206.

[26] Jin, R., Chen, W. and Sudjianto, A., 2002, "On Sequential Sampling for Global Metamodeling in Engineering Design", In *2002 ASME Design Engineering Technical Conference*, DETC2002/DAC-34092.

[27] Martin, J. D. and Simpson, T. W., 2002, "Use of Adaptive Metamodeling for Design Optimization", In *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, AIAA-2002-5631.