Performance study of Multi-Fidelity Gradient Enhanced Kriging

Selvakumar Ulaganathan · Ivo Couckuyt · Francesco Ferranti · Eric Laermans · Tom Dhaene

Received: date / Accepted: date

Abstract Multi-fidelity surrogate modelling offers an efficient way to approximate computationally expensive simulations. In particular, Kriging-based surrogate models are popular for approximating deterministic data. In this work, the performance of Kriging is investigated when multi-fidelity gradient data is introduced along with multi-fidelity function data to approximate computationally expensive black-box simulations. To achieve this, the recursive CoKriging formulation is extended by incorporating multi-fidelity gradient information. This approach, denoted by Gradient-Enhanced recursive CoKriging (GECoK), is initially applied to two analytical problems. As expected, results from the analytical benchmark problems show that additional gradient information of different fidelities can significantly improve the accuracy of the Kriging model. Moreover, GECoK provides a better approximation even when the gradient information is only partially available. Further comparison between CoKriging, Gradient Enhanced Kriging, denoted by GEK, and GECoK highlights various advantages of employing single and multi-fidelity gradient data. Finally, GECoK is further applied to two real-life examples.

Keywords Surrogate modelling · Gradient enhancement · Recursive CoKriging · Multi-fidelity modelling

S. Ulaganathan (Γ) · I. Couckuyt · F. Ferranti · E. Laermans · T. Dhaene
Ghent University - iMINDS, Department of Information Technology (INTEC), Gaston Crommenlaan 8, 9050 Ghent, Belgium
E-mail: selvakumar.ulaganathan@ugent.be

I. Couckuyt
E-mail: ivo.couckuyt@ugent.be

F. Ferranti
E-mail: francesco.ferranti@ugent.be

E. Laermans
E-mail: eric.laermans@ugent.be

T. Dhaene
E-mail: tom.dhaene@ugent.be

1 Introduction

The computational complexity of physics-based simulation codes has grown phenomenally in recent years. Despite continual advancement in computing power, there is a growing reluctance among researchers in using high-fidelity analysis codes due to prohibitive computational cost. It is rarely feasible to make many repetitive runs of high-fidelity computer simulations. This complexity is more evident in routine activities, such as optimization, sensitivity analysis, design space exploration, etc. (Forrester et al. 2006). Faced with these limitations, the alternative of using approximations or surrogate models of the actual, computationally expensive analysis codes (surrogate modelling) has received critical acclaim in recent years. The intention of surrogate modelling is to accurately imitate the behaviour of the computation-intensive simulator over an entire input space with a minimal number of expensive simulations.

Jin et al. (2000), Simpson et al. (2001) and Wang and Shan (2006) provided an overview of the most commonly used surrogate modelling techniques. Kriging, which was proposed by Sacks et al. (1989b) for the design and analysis of computer experiments, is arguably popular to approximate deterministic data resulting from computer codes (Sacks et al. 1989a; Kleijnen 2009). One of the primary goals of surrogate modelling is to enhance the accuracy of surrogate mod-

els with additional data which are not computation-intensive. To this end, surrogate modelling by incorporating all the cheaply available additional information, such as gradients, Hessian data, multi-fidelity data, prior knowledge, etc., is increasingly popular (Forrester et al. 2008; Courrier et al. 2014). For example, Morris et al. (1993) proposed gradient enhancement in Kriging, known as direct Gradient Enhanced Kriging (GEK), and showed that GEK significantly reduces the training sample data required to provide accurate surrogate models. Chung and Alonso (2002) compared direct GEK with an alternative formulation of GEK, called indirect GEK, which uses the same mathematical formulation of Kriging, but augments the training data with additional function values estimated from the gradient information. The authors stated that indirect GEK is prone to numerical errors introduced during the estimation of additional function values from gradients, whereas direct GEK exhibits formulation complexity at high dimensionality. Liu (2003) further investigated indirect GEK and proposed an alternative approach based on Neural Networks trained with both function and gradient information, but its performance is lower than indirect GEK. Laurenceau and Sagaut (2008) studied an aerodynamic problem with direct and indirect formulations of GEK and concluded that indirect GEK outperforms direct GEK irrespective of their equivalent mathematical formulations. More recently, Laurent et al. (2013) applied direct GEK to mechanical functions which lead to significant reduction in number of training samples and computational cost required to construct accurate approximations as compared to Kriging without gradients.

In problems where function data can be obtained with various accuracy levels, coupling data of different fidelities can further reduce the required training data while providing more accurate approximations of the actual function. Obtaining variable-fidelity data is more popular in areas where the computational complexity of analysis codes is dominant such as computational fluid dynamics (CFD) and finite element (FE) analysis. For instance, gradient data of different fidelities can be cheaply obtained in FE/CFD applications with the use of perturbation analysis or adjoint and automatic differentiation tools. Brezillon and Dwight (2005) solved the adjoint of the Navier-Stokes equations using an in-house CFD code which consumes only approximately 10% of the computational time required for a single complete non-linear flow calculation to provide all the gradient information at a single sample point. Dwight and Han (2009), Laurenceau et al. (2010) and Chung and Alonso (2002) demonstrated that with adjoint formulation k-dimensional gradients can be estimated at a computational cost of estimating 1-2 function values at a single sample point. Schneider (2012) introduced a software package called FEINS in which the gradients can be estimated cheaply for FE problems. Degroote et al. (2013) recently demonstrated an efficient method of estimating gradients cheaply in the context of fluid-structure interaction (FSI) problems. In addition, a set of CFD methods with varying degrees of computational complexity (Panel theory, Euler equations, and Navier-Stokes equations) can be used to generate data of various accuracy levels. Further, a single physics-based simulation model, which can be evaluated on meshes of varying refinement, can also result in data of various accuracy levels. Furthermore, the results from partially converged simulations using a single physics-based simulation code can also be used to produce data of various degrees of accuracy. Coupling different fidelities of data, which results in accurate model representation of the actual simulator, is not an entirely new idea in surrogate modelling. For example, various multi-fidelity surrogate modelling methods are already presented in the literature (Craig et al. 1998; Cumming and Goldstein 2009; Goldstein and Wooff 2007; Higdon et al. 2004; Kennedy and O'Hagan 2000; Forrester et al. 2007; Bandler et al. 1994).

Craig et al. (1998) initially presented a linear regression formulation based multi-fidelity surrogate modelling which is further improved by Cumming and Goldstein (2009) using a Bayes linear formulation (Goldstein and Wooff 2007). An autoregressive formulation based multi-fidelity modelling is introduced by Kennedy and O'Hagan (2000) and is quoted to be very efficient by Forrester et al. (2007) and Qian and Wu (2008). Huang et al. (2006) proposed an optimisation procedure, denoted by multiple-fidelity sequential kriging optimisation (MFSKO), based on this autoregressive formulation. Leary et al. (2004) demonstrated a multi-fidelity Kriging model enhanced with two different fidelities of gradient data in an optimisation context. The authors observed that the gradient incorporated multi-fidelity Kriging model is more accurate than the standard multi-fidelity Kriging model without gradients. The authors further stated that the advantage of finding global optima with gradient incorporated multi-fidelity Kriging model over the standard multi-fidelity Kriging model without gradients is still ambiguous and unclear. Recently, Le Gratiet (2012) proposed a recursive formulation based multi-fidelity Kriging modelling. This approach significantly reduces the modelling complexity associated with multi-fidelity Kriging models. This is also the context of the work presented here. Additionally, we introduce the gradient-enhanced recursive CoKriging (GECoK) method, an extension of the

recursive CoKriging method (Le Gratiet 2012), which incorporates function and gradient data of multiple fidelities while constructing approximation models. We mainly focus on investigating the effect of multi-fidelity gradient enhancement in Kriging-based surrogate models. To that end, the analytical expressions for the gradients of various correlation functions with respect to design variables are derived, and the GECoK methodology is also investigated if only a part of the gradient information is available. Furthermore, the analytical equation of the likelihood gradients is derived and integrated in the GECoK formulation, by evaluating the gradients of various correlation functions with respect to the hyper-parameters. This investigation is carried out by applying GECoK to two analytical and two real-life examples.

2 Mathematical formulation

The Kriging prediction $\hat{y}(\mathbf{x}^*)$ of a function $y(\mathbf{x})$ at a prediction point $\mathbf{x}^*$ is built from a constant trend function, $\hat{\mu}$ and the realization of a stationary Gaussian random process which represents the local features of $y(\mathbf{x})$ around n sample points, $\mathbf{X} = \{\mathbf{x}^{(1)}, ..., \mathbf{x}^{(n)}\}^T$, as (Sacks et al. 1989b),

$$\hat{y}(\mathbf{x}^*) = \hat{\mu} + \boldsymbol{\psi}^T \boldsymbol{\Psi}^{-1}(\mathbf{y} - \mathbf{f}\hat{\mu}), \tag{1}$$

where $\boldsymbol{\psi}$ contains correlations between $\mathbf{x}^*$ and the sample data points; $\boldsymbol{\Psi}$ is a $n \times n$ symmetric matrix of correlation between the sample data points; $\mathbf{y}$ contains the function values of the sample data; $\mathbf{f}$ is a $n \times 1$ column vector of ones; and the trend function $\hat{\mu} = (\mathbf{f}^T \boldsymbol{\Psi}^{-1} \mathbf{f})^{-1} \mathbf{f}^T \boldsymbol{\Psi}^{-1} \mathbf{y}$.

The Maximum Likelihood Estimate (MLE) of the hyper-parameters $(\theta_m, m = 1, ..., k)$ is obtained by maximizing the concentrated ln-likelihood

$$\phi = \frac{-n\ ln(\hat{\sigma}^2) - ln|\boldsymbol{\Psi}|}{2}, \tag{2}$$

where k is the dimensionality, i.e., number of design variables and $\hat{\sigma}^2 = (\mathbf{y} - \mathbf{f}\hat{\mu})^T \boldsymbol{\Psi}^{-1} (\mathbf{y} - \mathbf{f}\hat{\mu})/n$ is the estimated Kriging variance. For details on the maximum likelihood estimation method in standard Kriging and GEK, the readers are referred to Davis and Morris (1997), Zimmermann (2010), March et al. (2010) and Zimmermann (2013).

Gradient Enhanced Kriging (GEK) is a natural extension of Kriging and incorporates additional gradient information of the sample data in building surrogate models. GEK mathematically varies from Kriging in terms of $\boldsymbol{\psi}$, $\boldsymbol{\Psi}$, $\mathbf{y}$ and $\mathbf{f}$. Hence, the GEK prediction $\hat{\dot{y}}^{GEK}(\mathbf{x}^*)$ of the function $y(\mathbf{x})$ becomes,

$$\hat{\dot{y}}^{GEK}(\mathbf{x}^*) = \hat{\mu} + \dot{\boldsymbol{\psi}}^T \dot{\boldsymbol{\Psi}}^{-1}(\dot{\mathbf{y}} - \dot{\mathbf{f}}\hat{\mu}), \tag{3}$$

where

$$\dot{\boldsymbol{\Psi}} = \begin{Bmatrix} \boldsymbol{\Psi} & \frac{\partial \boldsymbol{\Psi}}{\partial x_1^{(i)}} & \cdots & \frac{\partial \boldsymbol{\Psi}}{\partial x_v^{(i)}} & \cdots & \frac{\partial \boldsymbol{\Psi}}{\partial x_k^{(i)}} \\ \frac{\partial \boldsymbol{\Psi}}{\partial x_1^{(j)}} & \frac{\partial^2 \boldsymbol{\Psi}}{\partial x_1^{(i)} \partial x_1^{(j)}} & \cdots & \frac{\partial^2 \boldsymbol{\Psi}}{\partial x_1^{(i)} \partial x_v^{(j)}} & \cdots & \frac{\partial^2 \boldsymbol{\Psi}}{\partial x_1^{(i)} \partial x_k^{(j)}} \\ \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ \frac{\partial \boldsymbol{\Psi}}{\partial x_u^{(j)}} & \frac{\partial^2 \boldsymbol{\Psi}}{\partial x_1^{(j)} \partial x_u^{(i)}} & \cdots & \frac{\partial^2 \boldsymbol{\Psi}}{\partial x_u^{(i)} \partial x_v^{(j)}} & \cdots & \frac{\partial^2 \boldsymbol{\Psi}}{\partial x_u^{(i)} \partial x_k^{(j)}} \\ \vdots & \vdots & \cdots & \vdots & \ddots & \vdots \\ \frac{\partial \boldsymbol{\Psi}}{\partial x_k^{(j)}} & \frac{\partial^2 \boldsymbol{\Psi}}{\partial x_1^{(j)} \partial x_k^{(i)}} & \cdots & \frac{\partial^2 \boldsymbol{\Psi}}{\partial x_v^{(j)} \partial x_k^{(i)}} & \cdots & \frac{\partial^2 \boldsymbol{\Psi}}{\partial x_k^{(i)} \partial x_k^{(j)}} \end{Bmatrix}, \tag{4}$$

$$\dot{\boldsymbol{\psi}} = \left(\boldsymbol{\psi}, \frac{\partial \boldsymbol{\psi}}{\partial x_1}, ..., \frac{\partial \boldsymbol{\psi}}{\partial x_k} \right)^T, \tag{5}$$

$$\dot{\mathbf{y}} = \left(\mathbf{y}, \frac{\partial \mathbf{y}}{\partial x_1}, ..., \frac{\partial \mathbf{y}}{\partial x_k} \right)^T, \tag{6}$$

$$\dot{\mathbf{f}} = \left(1_1, ...1_n, 0_{n+1}, ..., 0_{(k+1)n} \right)^T, \tag{7}$$

where $\dot{\boldsymbol{\psi}}$ contains correlations of both function and gradient data between $\mathbf{x}^*$ and the sample data points, $\dot{\boldsymbol{\Psi}}$ is a $(k+1)n \times (k+1)n$ symmetric block matrix and contains the correlations of function and gradient data between the sample data points and $\dot{\mathbf{y}}$ contains the function and the gradient values of the sample data. The notations $\frac{\partial \boldsymbol{\Psi}}{\partial x_u^{(j)}}$ and $\frac{\partial^2 \boldsymbol{\Psi}}{\partial x_u^{(i)} \partial x_v^{(j)}}$ denote the correlation between function and u^{th} dimension gradients and correlation between u^{th} dimension gradients and v^{th} dimension gradients, respectively. The direction of differentiation is denoted by i and j with $x^{(i)}$ and $x^{(j)}$ denoting two different samples.

CoKriging is considered as a multi-response extension to Kriging whereas GECoK is a multi-response extension to GEK. GECoK exploits the relationship between low-fidelity and high-fidelity sample data to enhance the prediction accuracy. It is based on the recursive CoKriging model of Le Gratiet (2012). Additionally, it incorporates multi-fidelity gradient data along with multi-fidelity function data to enhance surrogate model accuracy. Although GECoK can be easily extended to multi sets of variable-fidelity data, we limit ourselves to two sets of variable-fidelity data, i.e., a low- and a high-fidelity data set. GECoK uses two Kriging models to approximate the high-fidelity and the low-fidelity sample data. These two Kriging models are further related by a scaling or regression parameter ρ. Hence, building a GECoK model can be interpreted as constructing two separate GEK models independently. The first GEK model $\hat{\dot{y}}_c^{GEK}(\mathbf{x})$ is constructed with the low-fidelity data $(\mathbf{X}_c, \dot{\mathbf{y}}_c)$, where

$$\dot{\mathbf{y}}_c = \left(\mathbf{y}_c, \frac{\partial \mathbf{y}_c}{\partial x_1}, ..., \frac{\partial \mathbf{y}_c}{\partial x_k} \right)^T. \tag{8}$$

The hyper-parameters $\boldsymbol{\theta}_c$ for the low-fidelity GEK model can be found by maximizing the concentrated ln-likelihood

$$\phi_c^{GEK} = \frac{-(n_c(k_c'+1))\ ln(\hat{\sigma}_c^{2^{GEK}}) - ln|\dot{\boldsymbol{\Psi}}_c|}{2}, \tag{9}$$

where $k_c' \leq k$ is the number of dimensions in which the (low-fidelity) set of gradients is incorporated and $\hat{\sigma}_c^{2^{GEK}} = (\dot{\mathbf{y}}_c - \dot{\mathbf{f}}\hat{\dot{\mu}}_c)^T \dot{\boldsymbol{\Psi}}_c^{-1}(\dot{\mathbf{y}}_c - \dot{\mathbf{f}}\hat{\dot{\mu}}_c)/(n_c(k_c'+1))$ is the estimated variance of the low-fidelity GEK model.

Based on the method of ρ estimation, there are two different possible ways of constructing the second GEK model. Traditionally, ρ is estimated using MLE. In that case, the second GEK model is constructed with the residuals of the scaled low-fidelity data and high-fidelity data ($\mathbf{X}_e$, $\dot{\mathbf{y}}_d = \dot{\mathbf{y}}_e - \rho\hat{\dot{\mathbf{y}}}_c^{GEK}(\mathbf{X}_e)$), where

$$\dot{\mathbf{y}}_e = \left(\mathbf{y}_e, \frac{\partial \mathbf{y}_e}{\partial x_1}, ..., \frac{\partial \mathbf{y}_e}{\partial x_k}\right)^T, \tag{10}$$

$$\dot{\mathbf{y}}_d = \left(\mathbf{y}_e - \rho\hat{\mathbf{y}}_c^{GEK}, \frac{\partial \mathbf{y}_e}{\partial x_1} - \rho\frac{\partial \hat{\mathbf{y}}_c^{GEK}}{\partial x_1}, ..., \frac{\partial \mathbf{y}_e}{\partial x_k} - \rho\frac{\partial \hat{\mathbf{y}}_c^{GEK}}{\partial x_k}\right)^T \tag{11}$$

and $\hat{\mathbf{y}}_c^{GEK}(\mathbf{X}_e)$ is the GEK estimate from the low-fidelity model. Here the notation $\hat{\mathbf{y}}_c^{GEK}$ denotes the low-fidelity GEK estimate of the response values (not gradients) only. The low-fidelity GEK estimate is only required when there is no low-fidelity data $\dot{\mathbf{y}}_c$ available at $\mathbf{X}_e$. The hyper-parameters $\boldsymbol{\theta}_d$ for the second GEK model along with ρ can be similarly found by maximizing the concentrated ln-likelihood (see Equation 9) with the residual data ($\mathbf{X}_e$, $\dot{\mathbf{y}}_d$). With all the hyper-parameters $\boldsymbol{\theta}_c$ and $\boldsymbol{\theta}_d$ estimated, the resulting recursive GECoK interpolant $\hat{y}_e^{REC}$ of the high-fidelity data can then be defined as,

$$\hat{y}_e^{REC}(\mathbf{x}^*) = \rho\hat{y}_c^{GEK}(\mathbf{x}^*) + \hat{y}_d^{GEK}(\mathbf{x}^*). \tag{12}$$

It is also possible to calculate ρ using a least squares formulation as shown by Le Gratiet (2012). The advantage of using this formulation is that ρ can be expressed as a function of $\mathbf{X}$ which may provide more accurate estimation for ρ as shown in Le Gratiet (2012). This leads to an alternative formulation where the second GEK model is directly built with high-fidelity data ($\mathbf{X}_e$, $\dot{\mathbf{y}}_e$) and ρ is estimated from a least squares formulation as,

$$\rho_c = \frac{H_e^T\left(\frac{\dot{\boldsymbol{\Psi}}_e(X_e,X_e)^{-1}}{\sigma_e^2}\right)\dot{y}_e(\mathbf{X_e})}{H_e^T\left(\frac{\dot{\boldsymbol{\Psi}}_e(X_e,X_e)^{-1}}{\sigma_e^2}\right)H_e}, \tag{13}$$

where $H_e = G_c \cdot \dot{y}_c(\mathbf{X_e})\mathbf{1}^T\mathbf{f}$, σ_e^2 is the estimated Kriging variance for the second GEK model and G_c is a vector which contains the values of the regression function in $\mathbf{X_e}$. The recursive GECoK interpolant of the high-fidelity data for the latter formulation is then expressed as,

$$\hat{\dot{y}}_e^{REC}(\mathbf{x}^*) = \rho\hat{\dot{y}}_c^{GEK}(\mathbf{x}^*) + \hat{\dot{\mu}}_e + \dot{\boldsymbol{\psi}}_e^T\dot{\boldsymbol{\Psi}}_e^{-1}(\dot{\mathbf{y}}_e - \rho\hat{\dot{\mathbf{y}}}_c^{GEK}(\mathbf{X}_e) - \mathbf{f}\hat{\dot{\mu}}_e). \tag{14}$$

Note that the second part of the Equation 14 is similar to the GEK prediction formula (see Equation 3), except the fact that the interpolation is now on $\dot{\mathbf{y}}_d$ while, in Equation 12, $\dot{\mathbf{y}}_d$ is calculated upfront before the fitting of the second GEK model.

The Cholesky decomposition is normally used to factorize $\dot{\boldsymbol{\Psi}}$ during the estimation of hyper-parameters which involves a k-dimensional non-linear optimization. The Cholesky decomposition is the most time consuming part of Kriging modelling, and it becomes even more computation-intensive (a computational cost of $O(((k+1)n)^3)$ and a memory cost of $O(((k+1)n)^2)$) in GEK due to additional $n \times k$ rows/columns of $\dot{\boldsymbol{\Psi}}$. In this work, the analytical gradients of Equation 2 are introduced during the likelihood optimization so that the number of likelihood evaluations can be reduced, which in turn reduces the overall computational cost of the likelihood optimization (Toal et al. 2009). The analytical gradients of Equation 2 with respect to $\boldsymbol{\theta}$ are defined as (Toal et al. 2009),

$$\frac{\partial \phi}{\partial \boldsymbol{\theta}} = \frac{1}{2\hat{\sigma}^2}\left[(\dot{\mathbf{y}} - \mathbf{f}\hat{\dot{\mu}})^T\dot{\boldsymbol{\Psi}}^{-1}\frac{\partial \dot{\boldsymbol{\Psi}}}{\partial \boldsymbol{\theta}}\dot{\boldsymbol{\Psi}}^{-1}(\dot{\mathbf{y}} - \mathbf{f}\hat{\dot{\mu}})\right] - \frac{1}{2}\left[\dot{\boldsymbol{\Psi}}^{-1}\frac{\partial \dot{\boldsymbol{\Psi}}}{\partial \boldsymbol{\theta}}\right]^T. \tag{15}$$

The derivatives of every element of the correlation matrix $\dot{\boldsymbol{\Psi}}$ with respect to $\boldsymbol{\theta}$ must be estimated to completely define the Equation 15. Various correlation functions, which can be different for each GEK model, can be employed (Näther and Šimák 2003; Rasmussen and Williams 2006; Stein 1999; Šimák 2002). As the correlation functions must be differentiated twice in GEK to provide the correlation between gradient observations, we limit ourselves to the popular Gaussian correlation function and one instance of Matérn class of correlation functions. The Gaussian correlation function is defined as,

$$\psi(d) = exp\left(-\sum_{m=1}^{k}\theta_m d_m^2\right), \tag{16}$$

where $d = |x_m^i - x_m^j|$. The Matérn $\frac{5}{2}$ correlation function, which is more widely used in the machine learning context, can be expressed as,

$$\psi_{\nu=\frac{5}{2}}(d) = (1 + \sqrt{5}a + \frac{5a^2}{3})exp\left(-\sqrt{5}a\right), \tag{17}$$

where $a = \sqrt{\sum_{m=1}^{k} \theta_m d_m^2}$. The gradient, Hessian and their derivatives with respect to θ_k of the Gaussian and the Matérn $\frac{5}{2}$ correlation functions are given in Appendix A.

3 Test problems

Two analytical examples and two simulation examples are used as test problems. Normalized Root Mean Square Error (NRMSE), R Squared Error (R^2), Relative Average Absolute Error (RAAE) and Relative Maximum Absolute Error (RMAE) on a validation data set of n_p uniformly distributed pseudorandom points are used as error metrics to assess the surrogate model accuracy. The error metrics are expressed as,

$$NRMSE = \frac{\sqrt{\frac{\sum_{i=1}^{n_p}\left(y_t^i - \hat{y}^{REC^i}\right)^2}{n_p}}}{max(\boldsymbol{y}_t) - min(\boldsymbol{y}_t)}, \tag{18}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n_p}\left(y_t^i - \hat{y}^{REC^i}\right)^2}{\sum_{i=1}^{n_p}\left(y_t^i - \overline{y}_t\right)^2}, \tag{19}$$

$$RAAE = \frac{\sum_{i=1}^{n_p}|y_t^i - \hat{y}^{REC^i}|}{std(\boldsymbol{y}_t)n_p}, \tag{20}$$

$$RMAE = \frac{1}{std(\boldsymbol{y}_t)}\left(max(|y_t^1 - \hat{y}^{REC^1}|, |y_t^2 - \hat{y}^{REC^2}|, \ldots, |y_t^{n_p} - \hat{y}^{REC^{n_p}}|)\right), \tag{21}$$

where $\boldsymbol{y}_t$ is the vector of true response values, $\hat{\boldsymbol{y}}^{REC}$ is the vector of predicted response values, $\overline{y}$ is the mean of the true response values on n_p sample points and *std* stands for standard deviation. NRMSE, R^2 and RAAE show the overall surrogate modelling accuracy whereas RMAE is a local error metric. The values of NRMSE and RAAE approach zero as the overall surrogate model accuracy increases whereas high and small values are preferable for R^2 and RMAE, respectively.

3.1 Analytical examples

The one-dimensional analytical function is obtained from Forrester et al. (2008). Its expensive and cheap versions are defined as,

$$f_e(x) = (6x - 2.0)^2 sin(12x - 4.0), x \in [0, 1] \tag{22}$$

and

$$f_c(x) = 0.5(f_e(x)) + 10(x - 0.5) - 5, \tag{23}$$

respectively. The two-dimensional Peaks function is a built-in MATLAB function. The expensive and cheap versions of the Peaks function are defined as,

$$\begin{aligned} f_e(x,y) = {} & 3(1-x)^2 exp(-x^2 - (y+1)^2) \\ & - 10(\frac{x}{5} - x^3 - y^5)exp(-x^2 - y^2) \\ & - \frac{exp(-(x+1)^2 - y^2)}{3}, \quad [x,y] \in [-3,3] \end{aligned} \tag{24}$$

and

$$f_c(x,y) = 0.95 f_e(x',y'), \qquad x' = 0.97x \;\&\; y' = 1.05y, \tag{25}$$

respectively.

3.2 Simulation examples

In the first simulation example, a three turn spiral inductor (see Figure 1) is analyzed. The width of the conductors W and the outer length D_{out} are considered as design variables and their corresponding ranges are shown in Table 1. GECoK is used to model the quality factor Q and the inductance L of the spiral inductor at the frequency $f = 2.4$ GHz. The quality factor and the inductance can be expressed as (Yu and Bandler 2006),

$$Q(f, W, D_{out}) = -\frac{\Im m(Y_{11}(f, W, D_{out}))}{\Re e(Y_{11}(f, W, D_{out}))} \tag{26}$$

and

$$L(f, W, D_{out}) = -\frac{1}{2\pi f}\Im m\left(\frac{1}{Y_{12}(f, W, D_{out})}\right), \tag{27}$$

where the admittance parameters (Y-parameters) are simulated using the full-wave electromagnetic simulator Agilent Advanced Design System[1] (ADS) Momentum. The low-fidelity data are obtained using circuit schematic ADS simulations, while the high-fidelity data are obtained using full-wave electromagnetic ADS Momentum simulations. The circuit schematic simulations

[1] *www.eesof.com*, Agilent Technologies EEsof EDA, Santa Rosa, CA.

are based on analytical formulas for the electrical behavior of the spiral inductor, while the electromagnetic simulations are based on the solution of Maxwell's equations and are computationally more expensive. Unfortunately, ADS Momentum does not provide gradient information and therefore a parametric modeling approach (Chemmangat et al. 2011) is used to generate macromodels to compute low-fidelity and high-fidelity gradient data. A set of low-fidelity and a set of high-fidelity Y-parameters are collected on a grid of $51\times7\times7$ (f, W, D_{out}) samples over the frequency range $[0-5]$ GHz to build these parametric models that are then used to generate low-fidelity and high-fidelity gradient data. The computational time to get low-fidelity and high-fidelity admittance parameters at one sample point in the 2D (W, D_{out}) design space is roughly equal to $0.0301s$ and $0.2541s$, respectively, using a contemporary Windows desktop with Intel Core i3-2310M CPU @ 2.10GHz processor and 4GB RAM. The computational time to acquire one set of 2-dimensional gradient data of Q and L is very similar for both low-fidelity and high-fidelity cases as both are estimated from the very similar macromodels and is roughly equal to $0.0013s$. The second simulation example is a microwave inter-digital filter which is often used in cellular phones (Couckuyt et al. 2010) (see Figure 2). This filter has 5 geometrical parameters and the parameter ranges are given in Table 2. GECoK is used to model the mean of magnitude of the scattering S_{11} parameter of the filter, denoted by $|S_{11}|_{mean}$, in the frequency range $[2.2-2.6]$ GHz. The function and gradient data of S_{11} are obtained by carrying out full-wave electromagnetic simulations using CST[2] MicroWave Studio (CST MWS) at 5001 frequency samples for each instance simulation. The data is further used to obtain the function and gradient data of $|S_{11}|_{mean}$. Data of two different fidelities are obtained by carrying out full-wave electromagnetic simulations on two meshes of varying refinement (coarse and dense). The dense mesh has approximately 48000 tetrahedral cells and takes about $900s$ to provide a converged solution (i.e., high-fidelity data). The coarse mesh has a fixed number of 7823 tetrahedral cells and takes about $130s$ to provide a converged solution (i.e., low-fidelity data). The computational time of acquiring one function value is roughly equal to that of acquiring 10 and 20 sets of 5-dimensional gradients for the dense and coarse meshes, respectively.

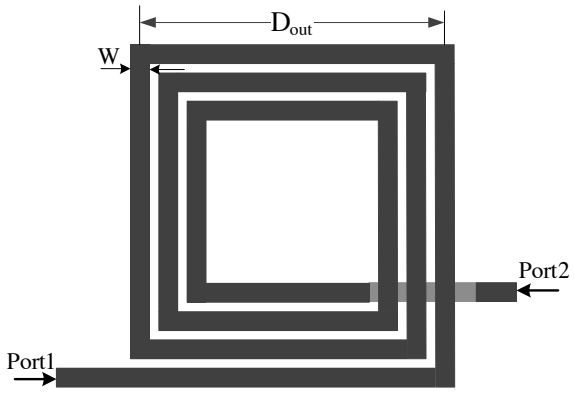
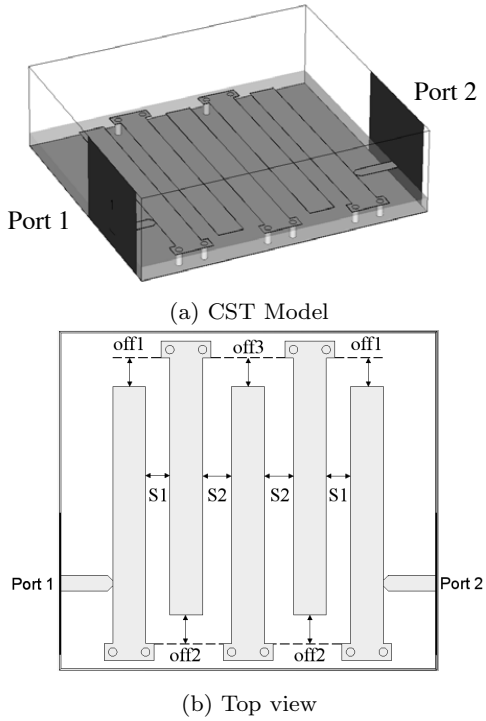


Fig. 1: Layout of the spiral inductor (Top View): The dielectric is 300 μm thick with a relative dielectric constant $\epsilon_r = 9.6$ and a loss tangent $tan\delta = 0.0002$. The conductivity of the metallic layers is equal to $\sigma = 5.8 \cdot 10^7$ S/m. The spacing between conductors is equal to 10 μm.

Table 1: Design parameters and their range of values. (Spiral inductor 2D)

Parameter	W	D_{out}
Lower Bound (in μm)	4	140
Upper Bound (in μm)	15	210



(a) CST Model

(b) Top view

Fig. 2: The 5 geometric parameters implicitly define the length of the microstrips ($off1$, $off2$ and $off3$) and the spacing between the microstrips ($S1$ and $S2$).

[2] *www.cst.com*, CST Computer Simulation Technology AG, Darmstadt, Germany.

Table 2: Design parameters and their range of values. (Microwave inter-digital filter 5D)

Parameter	$S1$	$S2$	$off1$	$off2$	$off3$
Lower Bound (in mm)	30.04	39.01	-10	-10	-10
Upper Bound (in mm)	40.04	49.01	10	10	10

4 Results and discussion

4.1 Analytical 1D test function

Figure 3 shows the high-fidelity and the low-fidelity one-dimensional functions along with their approximations. The design space is equidistantly sampled at 2 expensive and 7 cheap sample points. The Kriging approximation with the high-fidelity corner points gives a poor prediction whereas the GEK approximation results in a slightly better prediction than Kriging due to the inclusion of gradient information at $\mathbf{X}_e$. The recursive CoKriging results in better approximation than both Kriging and GEK approximations due to the incorporation of additional low-fidelity data, but the GECoK approximation lies exactly on f_e, being better than all the other Kriging models. It takes 4 expensive and 11 cheap data for CoKriging in order to overlay the GECoK approximation (see Figure 4).

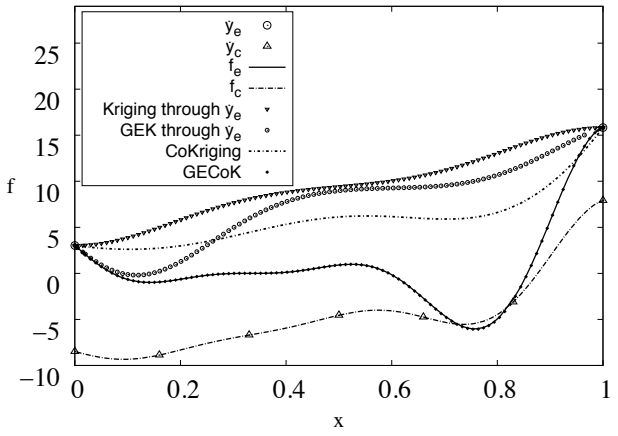


Fig. 3: Kriging with $n_e = 2$; GEK with $n_e = 2$ + gradients; CoKriging with $n_e = 2$ and $n_c = 7$ and GECoK with ($n_e = 2$ and $n_c = 7$) + gradients. All models are based on the Gaussian correlation function. (1D Function)

4.2 Analytical 2D test function

Figure 5 depicts the evolution of NRMSE as a function of n_e for the two-dimensional Peaks function. It can be

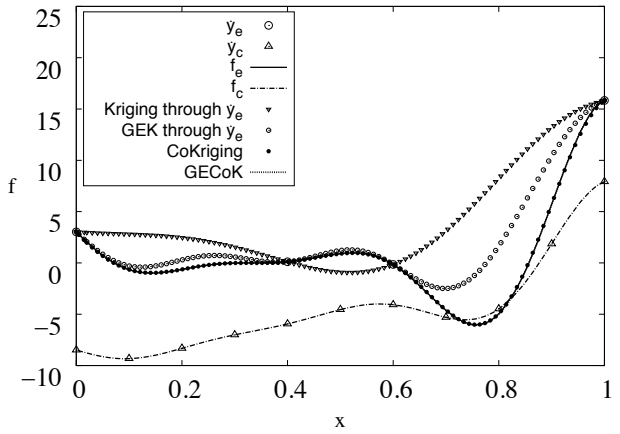


Fig. 4: Kriging with $n_e = 4$; GEK with $n_e = 4$ + gradients; CoKriging with $n_e = 4$ and $n_c = 11$ and GECoK with ($n_e = 2$ and $n_c = 7$) + gradients. All models are based on the Gaussian correlation function. (1D Function)

observed that GECoK models considerably reduce the number of expensive samples required, more than 55% in this case, to achieve the equivalent accuracy level of CoKriging models. This is clearly the effect of incorporating additional gradient information at X_e and X_c. Moreover, the Gaussian correlation function based GECoK model requires just 5 expensive and 9 cheap sample points to achieve 60% of accuracy improvement, in terms of NRMSE, over its corresponding CoKriging model (see Table 3). The Gaussian correlation function based CoKriging model fails to accurately describe the Peaks function with 5 expensive and 9 cheap sample points (see Figure 6), and requires 15 expensive and 38 cheap sample points to reach the equivalent accuracy level of GECoK. A similar behaviour is observed with the Matérn $\frac{5}{2}$ correlation function too (see Table 3). The inaccuracy of CoKriging models is caused by under-sampling and poor correlation among the available expensive sample points whereas GECoK benefits from the incorporation of additional gradient information at both $\mathbf{X}_e$ and $\mathbf{X}_c$. Moreover, histogram plots of the prediction error (difference between actual and modelled surface of the Peaks function) on a 50×50 uniform grid also compares in favour of GECoK models (see Figure 7).

The Gaussian correlation function based GEK model requires 9 expensive sample points to reach the accuracy level of GECoK with 5 expensive and 9 cheap points. GECoK reduces the computational burden of getting 4 expensive data with that of 9 cheap data in this case. In turn, it conveys that if cheap data (function + gradient) at more than $2 \times n_e$ sample points

Table 3: Error metrics on a validation data set of $n_p = 500$: Gaussian correlation function with $n_e = 5$ and $n_c = 9$ and Matérn 5/2 correlation function with $n_e = 9$ and $n_c = 30$. GEK uses only the expensive data. Imp. denotes *% of improvement* of GEK/GECoK models over their corresponding CoKriging models. (Peaks 2D)

			Gaussian					
Model	NRMSE	Imp.	R^2	Imp.	RAAE	Imp.	RMAE	Imp.
CoKriging	0.3453	-	0	-	0.8671	-	2.7004	-
GEK	0.4680	-36%	0	0%	0.9183	-6%	3.6398	-35%
GECoK	0.1377	60%	0.8271	>100%	0.3565	59%	0.6679	75%
			Matérn $\frac{5}{2}$					
CoKriging	0.1614	-	0.7627	-	0.3841	-	1.0646	-
GEK	0.2049	-27%	0.6176	-19%	0.5088	-32%	1.3358	-25%
GECoK	0.0445	72%	0.9820	29%	0.1035	73%	0.3740	65%

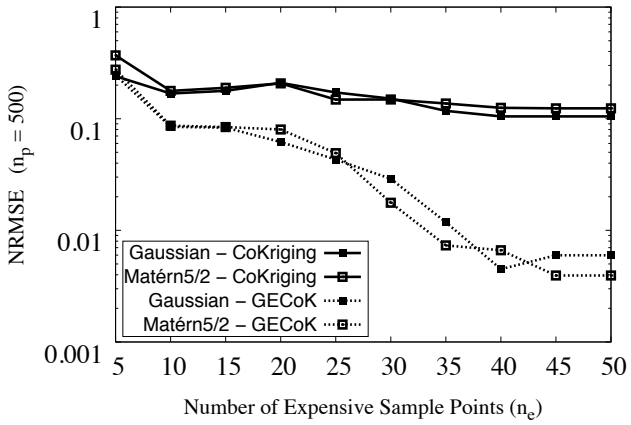
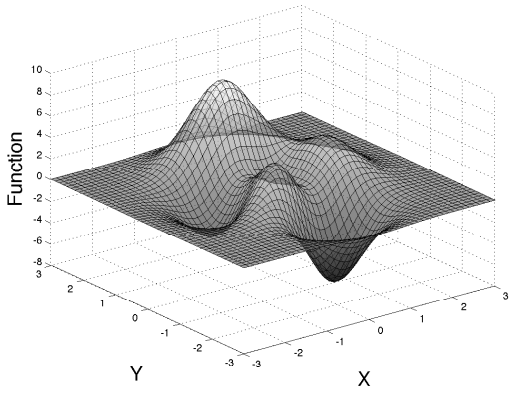


Fig. 5: Evolution of NRMSE on a validation data set of $n_p = 500$ for a varying number of expensive data. A constant number of cheap data $n_c = 50$ is used throughout for all the expensive runs. (Peaks 2D)

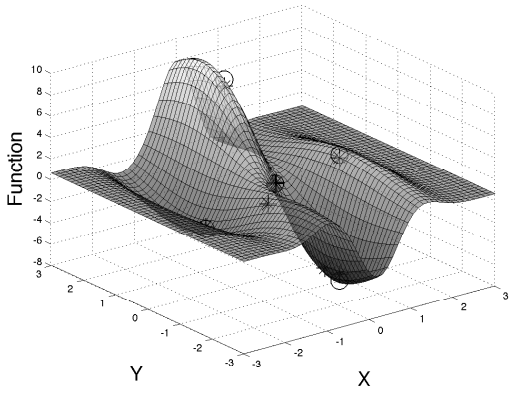
can be obtained in the computational cost of getting expensive data at n_e sample points, which is quite possible in many engineering problems, then GECoK can be utilized to achieve equally accurate approximations of GEK with less computational burden. Although this advantage, in general, is subject to the quality of the cheap data, the GECoK models essentially benefit from the abundantly available cheap 'function and gradient' data which help them to capture the correlation more realistically than GEK.

4.3 Simulation examples

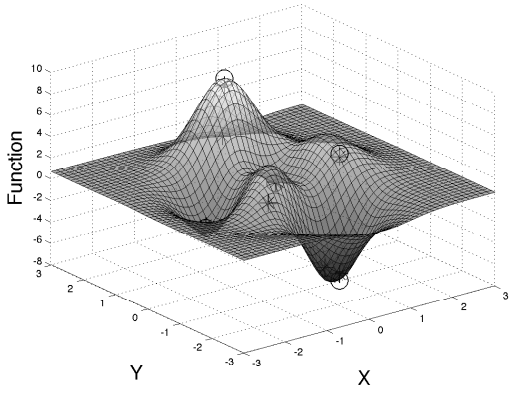
The simulation examples exhibit a similar accuracy improvement in GECoK models over the CoKriging models, as in the case of the analytical problems. For example, both the Gaussian and Matérn $\frac{5}{2}$ correlation functions are able to accurately model the functions, inductance L and quality factor Q, with just 3 expensive and 6 cheap sample points (see Figure 8); they are also able to achieve more than 50% of reduction in NRMSE than the CoKriging models (see Table 4) which require at least 6 (quality factor Q) to 13 (inductance L) expensive and 30 cheap sample points to reach the NRMSE achieved by the GECoK models. Surprisingly, GEK with $n_e = 3$ plus gradients results in more accurate models than CoKriging with $n_e = 3$ and $n_c = 6$ contrarily to the results of the analytical functions (see Tables 3 and 4). This is mainly due to the fact that the gradients in GEK restrict the possible interpolation through the function data and allows the likelihood optimization to successfully capture the sample data's covariance structure. But, in the case of CoKriging, the likelihood optimization struggles to capture the sample data's covariance structure because of the small number of samples in such a large design space and the likelihood function exhibits various local optima. This increases the probability of finding inaccurate hyperparameters in CoKriging which do not correspond to the maximum likelihood exist in the sample data's actual covariance structure. Moreover, as the likelihood function in CoKriging now exhibits multiple local optima, the optimizer becomes more vulnerable to the likelihood function behaviour and is more likely to converge at one of the local optima of the contour although it is very much based on the capability of the optimizer used (see Figure 9). In this work, a gradient-based optimizer called *fmincon* which is available in MATLAB is used. As the analytical gradients of the likelihood function are estimated (see Equation 15 and Appendix A) in this work, a gradient based solver can incorporate these gradients during the likelihood optimization. This reduces the number of likelihood function evaluations, which in turn reduces the overall surrogate modelling time. Whereas a more exhaustive search algorithm such as genetic algorithm may indeed find the global optimum but requires a greater computational cost. This result goes along with the intuitive fact that an expen-



(a) Actual



(b) CoKriging



(c) GECoK

Fig. 6: Actual function, Gaussian correlation function based CoKriging model with $n_e = 5$ and $n_c = 9$ and GECoK model with ($n_e = 5$ and $n_c = 9$) + gradients. The circle represents the HF sample points whereas the asterisk symbol represents the LF sample points. (Peaks 2D)

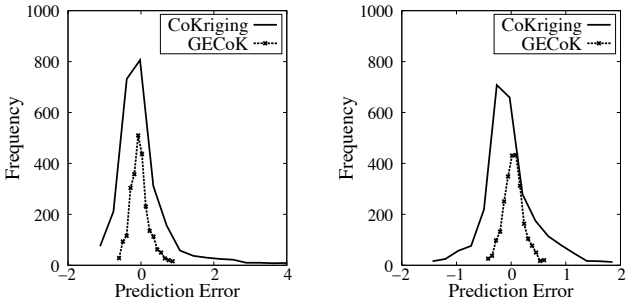


Fig. 7: Histogram of prediction error on a 50×50 uniform grid: Gaussian correlation function with $n_e = 10$ and $n_c = 19$ (left) and Matérn 5/2 correlation function with $n_e = 9$ and $n_c = 30$. (Peaks 2D)

sive gradient value is worth more than a cheap function value. A similar behaviour is exhibited during the modelling of $|S_{11}|_{mean}$ for the microwave inter-digital filter (see Table 6).

Among the correlation functions employed, the Gaussian correlation function based models are observed to be less accurate in most of the cases of the simulation examples (see Tables 4-6 and Figures 10-11). This result goes along with the fact that the Matérn class of correlation functions accounts for the negligible roughness present in data involving typical activities of numerical problem solving such as meshing and solver run (Rasmussen and Williams (2006)). Whereas the extreme smoothness requirement of the Gaussian correlation function is usually considered as unrealistic for real life data involving physical processes (Stein (1999)). The accuracy of GECoK models is also assessed by employing only a partial set of gradients. For example, the gradients can be incorporated either at X_c or X_e or even only in some of the dimensions of X_c and/or X_e subject to their availability. Employing only a partial set of gradients brings down the size of $\dot{\mathbf{\Psi}}$ to $(n + (n \times k')) \times (n + (n \times k'))$ with k' being the number of dimensions in which the partial set of gradients is incorporated. GECoK provides more accurate models than CoKriging even when the gradients are introduced only at X_e (see Figure 11). However, employing gradients only at X_c, in general, does not provide much advantage to GECoK models.

The surrogate modelling cost grows substantially in GECoK as the gradient information is now incorporated at both X_e and X_c (see Figure 12a). Hence, for a fair comparison, the surrogate modelling cost should be taken into account as well. In order to ensure an equal surrogate modelling cost for both CoKriging and GECoK models, the correlation matrix in CoKriging should be augmented with more function data, so that its size equals that of $\dot{\mathbf{\Psi}}$ in GECoK. This way CoKriging

Table 4: Error metrics on a validation data set of $n_p = 500$: Gaussian and Matérn 5/2 correlation functions with $n_e = 3$ and $n_c = 6$. GEK uses only the expensive data. Imp. denotes *% of improvement* of GEK/GECoK models over their corresponding CoKriging models. (Inductance 2D (L))

			Gaussian					
Model	NRMSE	Imp.	R^2	Imp.	RAAE	Imp.	RMAE	Imp.
CoKriging	1.71	-	0	-	3.7620	-	28.1353	-
GEK	0.0043	99%	0.9997	101%	0.0135	99%	0.0415	99%
GECoK	0.0045	99%	0.9996	101%	0.0141	99%	0.0436	99%
			Matérn $\frac{5}{2}$					
CoKriging	0.0271	-	0.9870	-	0.0965	-	0.2653	-
GEK	0.0082	69%	0.9988	1%	0.0250	74%	0.0852	67%
GECoK	0.0089	67%	0.9986	1%	0.0271	71%	0.0917	65%

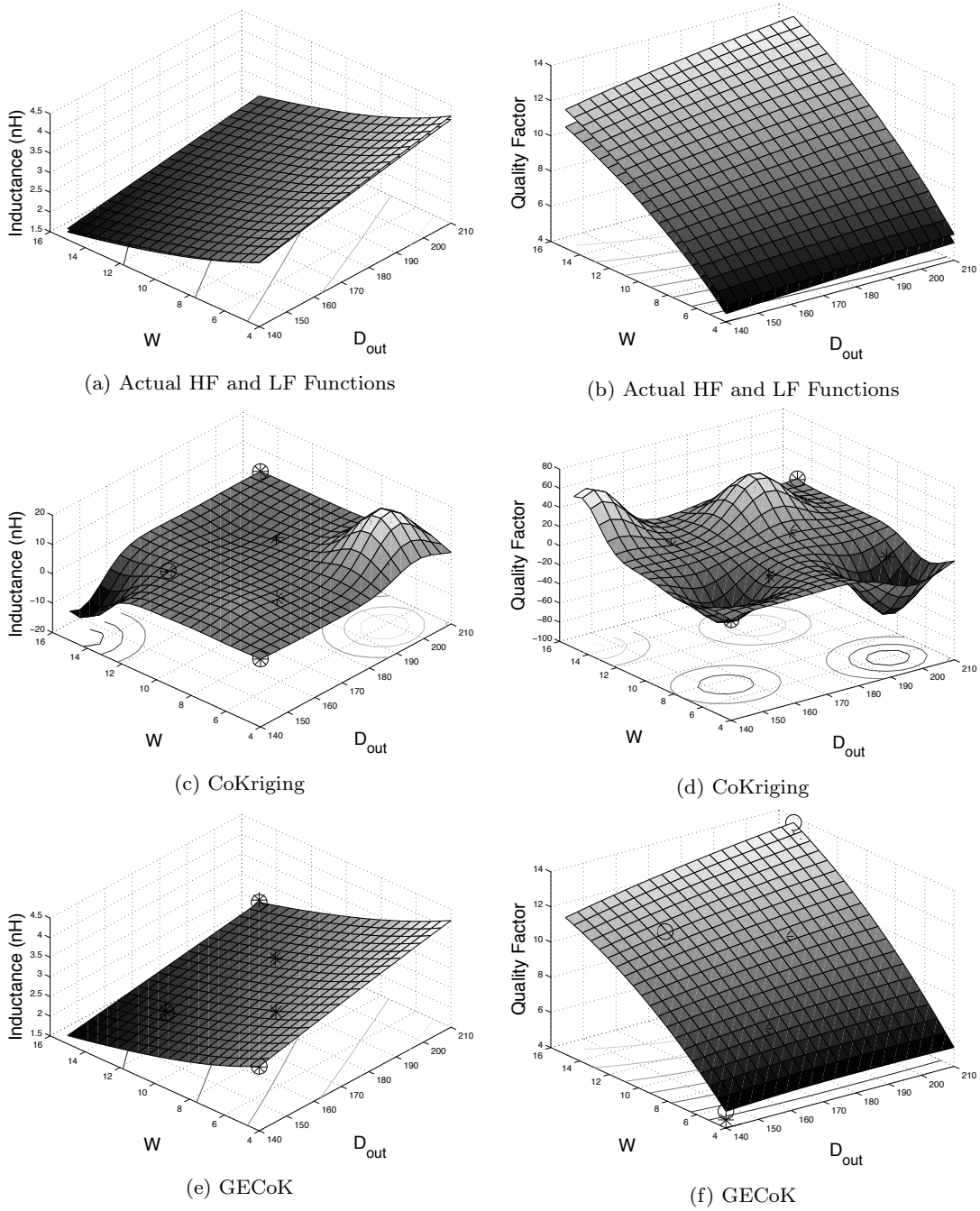
Table 5: Error metrics on a validation data set of $n_p = 500$: Gaussian and Matérn 5/2 correlation functions with $n_e = 3$ and $n_c = 6$. GEK uses only the expensive data. Imp. denotes *% of improvement* of GEK/GECoK models over their corresponding CoKriging models. (Quality factor 2D (Q))

			Gaussian					
Model	NRMSE	Imp.	R^2	Imp.	RAAE	Imp.	RMAE	Imp.
CoKriging	2.3015	-	0	-	5.5085	-	25.3232	-
GEK	0.0144	99%	0.9973	102%	0.0338	99%	0.1633	99%
GECoK	0.0152	99%	0.9971	102%	0.0359	99%	0.1688	99%
			Matérn $\frac{5}{2}$					
CoKriging	0.0165	-	0.99	-	0.0499	-	0.1225	-
GEK	0.0026	84%	0.9999	0.3%	0.0063	87%	0.0525	57%
GECoK	0.0025	85%	0.9999	0.3%	0.0063	87%	0.0411	66%

Table 6: Error metrics on a validation data set of $n_p = 50$: Gaussian and Matérn 5/2 correlation functions with $n_e = 50$ and $n_c = 100$. GEK uses only the expensive data. Imp. denotes *% of improvement* of GEK/GECoK models over their corresponding CoKriging models. ($|S_{11}|_{mean}$ 5D)

			Gaussian					
Model	NRMSE	Imp.	R^2	Imp.	RAAE	Imp.	RMAE	Imp.
CoKriging	0.2082	-	0.0057	-	0.8293	-	2.4620	-
GEK	0.0378	82%	0.9672	>100%	0.1281	85%	0.7110	71%
GECoK	0.0356	83%	0.9710	>100%	0.1217	85%	0.5687	77%
			Matérn $\frac{5}{2}$					
CoKriging	0.0524	-	0.9372	-	0.1852	-	0.8378	-
GEK	0.0324	38%	0.9760	4%	0.1070	42%	0.6414	23%
GECoK	0.0297	43%	0.9798	5%	0.0993	46%	0.4833	42%

and GECoK models can be compared subject to equal surrogate modelling cost as this cost is directly related to the Cholesky decomposition of $\dot{\mathbf{\Psi}}$. Results of the simulation examples show that CoKriging with augmented function data provides more accurate surrogate models than GECoK (see Figure 12b). This confirms the intuitive fact that a function value is worth more than a gradient value of equal fidelity. However, the computational cost of estimating additional function values for CoKriging, so that the size of $\mathbf{\Psi}$ equals that of $\dot{\mathbf{\Psi}}$, is significantly higher than that of acquiring function and gradient data for GECoK (see Figure 12c). Moreover, the surrogate modelling cost of GECoK models, in this case, is much lower than the computational cost spent on estimating additional function values for CoKriging models. These facts significantly tilt comparisons in favor of GECoK. As mentioned earlier in the Section 2, the scaling parameter ρ can be expressed as a function of $\mathbf{X}$ which results in more accurate surrogate models for a simulation-based example shown in Le Gratiet (2012) without gradients. However, when the gradient information is incorporated, expressing ρ as a function of X does not always lead to significant improvements in surrogate model accuracy (see Appendix B). Moreover, better ρ estimates are obtained, when it is calcu-



(a) Actual HF and LF Functions

(b) Actual HF and LF Functions

(c) CoKriging

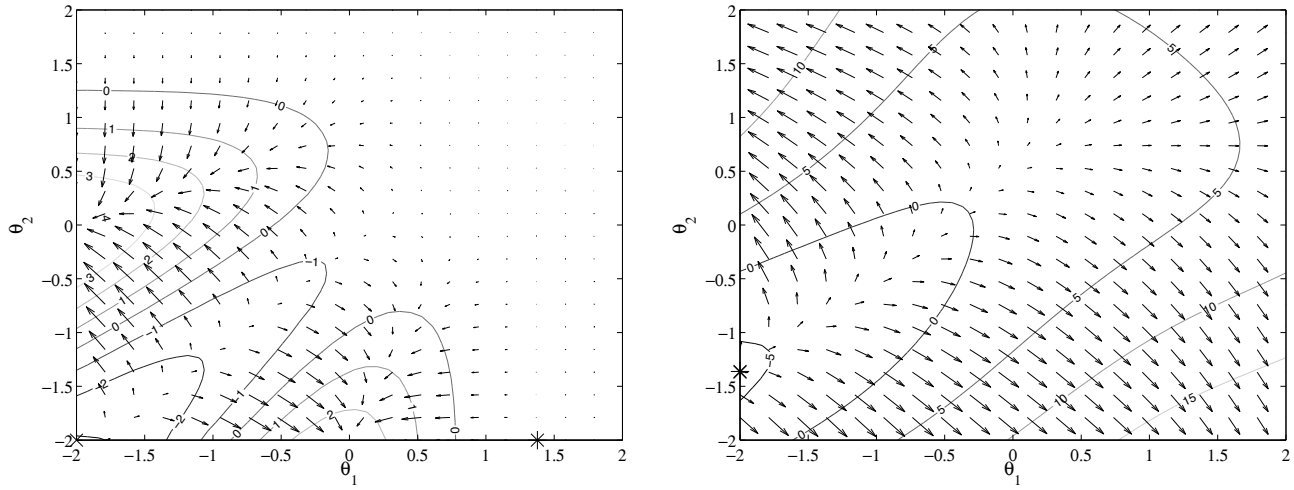
(d) CoKriging

(e) GECoK

(f) GECoK

Fig. 8: Actual HF and LF functions, Gaussian correlation function based CoKriging models with $n_e = 3$ and $n_c = 6$ and GECoK models with ($n_e = 3$ and $n_c = 6$) + gradients. The circle represents the HF sample points whereas the asterisk symbol represents the LF sample points. (Inductance 2D (Left side figures) and quality factor 2D)

lated via MLE, as it is now based on both X and the likelihood.



(a) CoKriging (b) GEK

Fig. 9: Contours of the likelihood function for the Matérn 5/2 correlation function based cheap CoKriging model with $n_c = 6$ and GEK model with $n_e = 3$ + gradients. The asterisk symbol denotes the location of optimal θ values found using MLE which maximize the likelihood function. The cross symbol denotes the location of optimal θ values found manually during the generation of these contour plots which maximize the likelihood function. The arrows denote the derivatives of the likelihood function. (Inductance 2D)

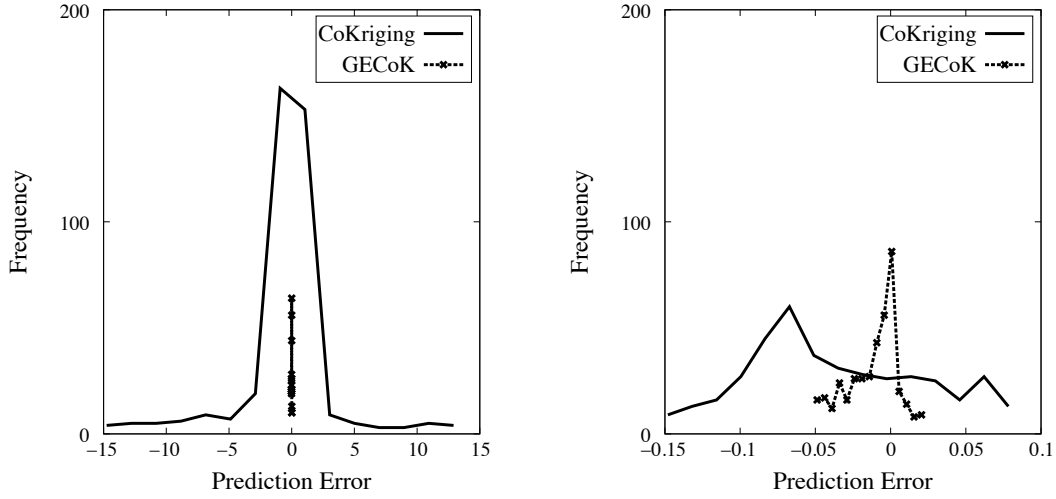
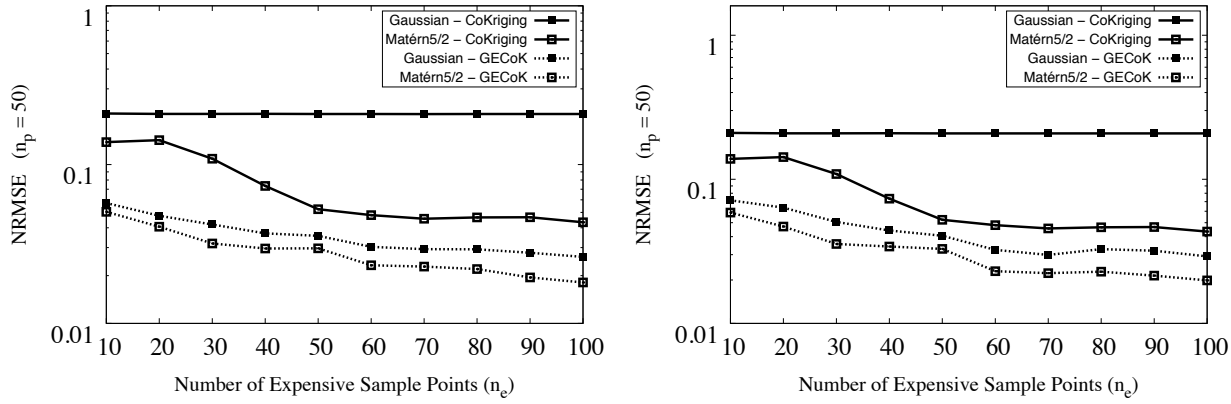


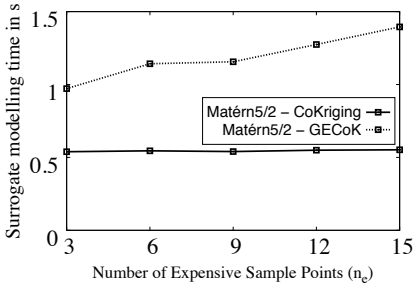
Fig. 10: Histogram of prediction error on a 20×20 uniform grid: Gaussian (left) and Matérn 5/2 correlation functions with $n_e = 3$ and $n_c = 6$. (Inductance 2D)



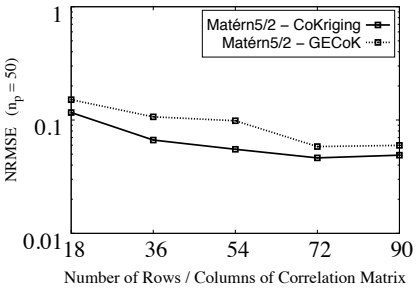
(a) Effect of expensive data

(b) GECoK without gradients at X_c

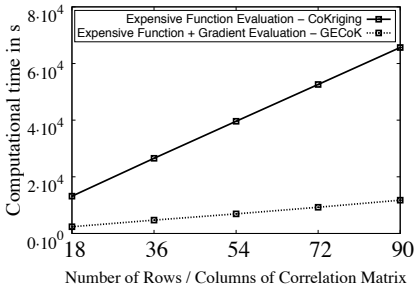
Fig. 11: Effect of a complete and a partial set of gradients. Evolution of NRMSE on a validation data set of $n_p = 50$ for a varying number of expensive data. A constant number of cheap data $n_c = 100$ is used throughout for all the expensive runs. ($|S_{11}|_{mean}$ 5D)



(a) Surrogate modelling cost of CoKriging and GECoK models.



(b) Efficiency of CoKriging models with $\boldsymbol{\Psi}$ being equal in size with $\dot{\boldsymbol{\Psi}}$ of GECoK models (i.e., equal surrogate modelling cost). CoKriging with $(n_e + k \times n_e)$ expensive function data and $(n_c + k \times n_c)$ cheap function data. Evolution of NRMSE on the validation data set for varying expensive data.



(c) Computational time spent on acquiring $n_e + (k \times n_e)$ expensive function data for CoKriging models and n_e expensive function data + $(k \times n_e)$ expensive gradient data for GECoK Models. The size of $\boldsymbol{\Psi}$ equals that of $\dot{\boldsymbol{\Psi}}$ (i.e., equal surrogate modelling cost).

Fig. 12: GECoK with n_e expensive function data + $(k \times n_e)$ expensive gradient data and n_c cheap function data + $(k \times n_c)$ cheap gradient data. The size of the cheap correlation matrix is fixed to 90×90 for both CoKriging and GECoK models. ($|S_{11}|_{mean}$ 5D)

5 Conclusions

This paper investigates the effects of multi-fidelity gradient enhancement in Kriging-based surrogate modelling. This has been carried out by introducing the Gradient-Enhanced recursive CoKriging (GECoK) approach which utilizes additional multi-fidelity gradient information to enhance the accuracy of Kriging. As expected, multi-fidelity gradient enhancement can significantly reduce the number of computationally expensive simulations required to provide accurate model representations. Moreover, results show that expensive gradient data often provides more information about the function to be modelled than cheap function data. This allows Gradient Enhanced Kriging (GEK) to result in more accurate model representations than CoKriging. This also allows GECoK to provide more accurate approximations than CoKriging with only a partial set of gradients. Further, this also reduces the size of the correlation matrix, which in turn improves the overall computational efficiency of GECoK modelling. However, function data is more informative than gradient data of equal fidelity. Hence, a CoKriging model with its correlation matrix size scaled up with more function data in order to equal that of GECoK outperforms GECoK; but, at a computational cost of estimating additional function data which is often significantly higher than that of estimating gradient data for GECoK. Furthermore, the cheap gradient data is rarely advantageous while it can increase the surrogate modelling cost significantly as it is often abundantly available. Hence, care should be taken when the cheap gradient data is incorporated in large quantities.

Acknowledgements

This research has been funded by the Interuniversity Attraction Poles Programme BESTCOM initiated by the Belgian Science Policy Office. Additionally, this research has been supported by the Fund for Scientific Research in Flanders (FWO-Vlaanderen). Ivo Couckuyt and Francesco Ferranti are post-doctoral research fellows of the Research Foundation Flanders (FWO-Vlaanderen). The authors like to thank Frank Mosler from Computer Simulation Technology (CST) for providing the microwave inter-digital filter example.

References

Bandler JW, Biernacki RM, Chen SH, Grobelny PA, Hemmers RH (1994) Space mapping technique for electromagnetic optimization. IEEE Transactions on Microwave Theory and Techniques 42(12):2536–2544

Brezillon J, Dwight R (2005) Discrete adjoint of the Navier-Stokes equations for aerodynamic shape optimization. In: Evolutionary and Deterministic Methods for Design, Optimisation and Control with Applications to Industrial and Societal Problems (EUROGEN 2005), Munich, Germany

Chemmangat K, Ferranti F, Knockaert L, Dhaene T (2011) Parametric macromodeling for sensitivity responses from tabulated data. IEEE Microwave and Wireless Components Letters 21(8):397–399

Chung HS, Alonso JJ (2002) Using gradients to construct cokriging approximation models for high-dimensional design optimization problems. In: Problems, 40th AIAA Aerospace Sciences Meeting and Exhibit, AIAA, Reno, NV, pp 2002–0317

Couckuyt I, Declercq F, Dhaene T, Rogier H, Knockaert L (2010) Surrogate-based infill optimization applied to electromagnetic problems. International Journal of RF and Microwave computer-aided Engineering 20(5):492–501

Courrier N, Boucard PA, Soulier B (2014) The use of partially converged simulations in building surrogate models. Advances in Engineering Software 67(0):186–197

Craig PS, Goldstein M, Seheult AH, Smith JA (1998) Constructing partial prior specifications for models of complex physical systems. Journal of the Royal Statistical Society Series D (The Statistician) 47(1):37–53

Cumming JA, Goldstein M (2009) Small sample Bayesian designs for complex high-dimensional models based on information gained using fast approximations. Technometrics 51(4):377–388

Davis GJ, Morris MD (1997) Six factors which affect the condition number of matrices associated with kriging. Mathematical Geology 29(5):669–683

Degroote J, Hojjat M, Stavropoulou E, Wüchner R, Bletzinger KU (2013) Partitioned solution of an unsteady adjoint for strongly coupled fluid-structure interactions and application to parameter identification of a one-dimensional problem. Structural and Multidisciplinary Optimization 47(1):77–94

Dwight RP, Han ZH (2009) Efficient uncertainty quantification using gradient-enhanced kriging. In: 11th AIAA Non-Deterministic Approaches Conference, Palm Springs, California, USA

Forrester AI, Bressloff NW, Keane AJ (2006) Optimization using surrogate models and partially converged computational fluid dynamics simulations. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 462(2071):2177–2204

Forrester AI, Sóbester A, Keane AJ (2007) Multi-fidelity optimization via surrogate modelling. Proceedings of the Royal Society 463:3251–3269

Forrester AI, Sóbester A, Keane AJ (2008) Engineering Design via Surrogate Modelling: A Practical Guide, 1st edn. Wiley

Goldstein M, Wooff DA (2007) Bayes Linear Statistics: Theory & Methods. Bayes Linear Statistics, Wiley

Higdon D, Kennedy M, Cavendish J, Cafeo J, Ryne RD (2004) Combining field data and computer simulations for calibration and prediction. SIAM Journal on Scientific Computing 26(2):448–466

Huang D, Allen T, Notz W, Miller R (2006) Sequential kriging optimization using multiple-fidelity evaluations. Structural and Multidisciplinary Optimization 32(5):369–382

Jin R, Chen W, Simpson TW (2000) Comparative studies of metamodeling techniques under multiple modeling criteria. Structural and Multidisciplinary Optimization 23:1–13

Kennedy MC, O'Hagan A (2000) Predicting the output from a complex computer code when fast approximations are available. Biometrika 87(1):1–13

Kleijnen J (2009) Kriging metamodeling in simulation: A review. European Journal of Operational Research 192(3):707–716

Laurenceau J, Sagaut P (2008) Building efficient response surfaces of aerodynamic functions with kriging and cokriging. AIAA 46(2):498–507

Laurenceau J, Meaux M, Montagnac M, Sagaut P (2010) Comparison of gradient-based and gradient-enhanced response-surface-based optimizers. American Institute of Aeronautics and Astronautics Journal 48(5):981–994

Laurent L, Boucard PA, Soulier B (2013) Generation of a cokriging metamodel using a multiparametric strategy. Computational Mechanics 51(2):151–169

Le Gratiet L (2012) Recursive co-kriging model for design of computer experiments with multiple levels of fidelity with an application to hydrodynamic. ArXiv e-prints 1210.0686

Leary SJ, Bhaskar A, Keane AJ (2004) A derivative based surrogate model for approximating and optimizing the output of an expensive computer simulation. Journal of Global Optimization 30(1):39–58

Liu W (2003) Development of gradient-enhanced kriging approximations for multidisciplinary design optimisation. PhD thesis, University of Notre Dame, Notre Dame, Indiana

March A, Willcox K, Wang Q (2010) Gradient-based multifidelity optimisation for aircraft design using bayesian model calibration. In: 2nd Aircraft Structural Design Conference, Royal Aeronautical Society, London, p 1720

Morris MD, Mitchell TJ, Ylvisaker D (1993) Bayesian design and analysis of computer experiments: Use of gradients in surface prediction. Technometrics 35(3):243–255

Näther W, Šimák J (2003) Effective observation of random processes using derivatives. Metrika Springer-Verlag 58:71–84

Qian PZG, Wu CFJ (2008) Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments. Technometrics 50(2):192–204

Rasmussen CE, Williams CKI (2006) Gaussian processes for machine learning. The MIT Press, Cambridge, MA, USA

Sacks J, Schiller SB, Welch WJ (1989a) Designs for computer experiments. Technometrics 31(1):41–47

Sacks J, Welch WJ, Mitchell TJ, Wynn HP (1989b) Design and analysis of computer experiments. Statistical Science 4(4):409–423

Schneider R (2012) Feins: Finite element solver for shape optimization with adjoint equations. In: Progress in industrial mathematics at ECMI 2010 Conference, pp 573–580

Simpson T, Poplinski J, Koch PN, Allen J (2001) Metamodels for computer-based engineering design: Survey and recommendations. Engineering with Computers 17(2):129–150

Stein ML (1999) Interpolation of Spatial Data: Some Theory for Kriging. Springer, New York

Toal DJ, Forrester AI, Bressloff NW, Keane AJ, Holden C (2009) An adjoint for likelihood maximization. Proc R Soc A 8 465(2111):3267–3287

Šimák J (2002) On experimental designs for derivative random fields. PhD thesis, TU Bergakademie Freiberg, Freiberg, Germany

Wang GG, Shan S (2006) Review of metamodeling techniques in support of engineering design optimization. Journal of Mechanical Design 129(4):370–380

Yu W, Bandler J (2006) Optimization of spiral inductor on silicon using space mapping. In: IEEE MTT-S International Microwave Symposium Digest, pp 1085–1088

Zimmermann R (2010) Asymptotic behavior of the likelihood function of covariance matrices of spatial gaussian processes. Journal of Applied Mathematics

Zimmermann R (2013) On the maximum likelihood training of gradient-enhanced spatial gaussian processes. SIAM Journal on Scientific Computing

35(6):A2554–A2574

A Analytical expressions for gradient, Hessian and likelihood gradients of correlation functions

A.1 Gaussian correlation function:

Gradient of correlation function with respect to X (i.e., cross-correlation):

$$\frac{\partial \mathbf{\Psi}^{(i,j)}}{\partial x^{(j)}} = 2\theta d \mathbf{\Psi}^{(i,j)} \tag{28}$$

Hessian of correlation function with respect to X (i.e., cross-correlation):

$$\frac{\partial^2 \mathbf{\Psi}^{(i,j)}}{\partial x_u^{(i)} \partial x_v^{(j)}} = \begin{cases} -4\theta_u \theta_v d_u d_v \mathbf{\Psi}^{(i,j)} & \text{if } u \neq v \\ [2\theta - 4\theta^2 d^2] \mathbf{\Psi}^{(i,j)} & \text{if } u = v \end{cases} \tag{29}$$

Derivative of correlation function with respect to θ_k:

$$\frac{\partial}{\partial \theta_k} \left(\psi(d_{u(v)}) \right) = -10^{\theta_{u(v)}} d_{u(v)}^2 log(10) exp\left(- \sum_{m=1}^{k} \theta_m d_m^2 \right) \tag{30}$$

Derivatives of cross-correlation functions with respect to θ_k:

$$\frac{\partial}{\partial \theta_k} \left(\frac{\partial \mathbf{\Psi}^{(i,j)}}{\partial x_v^{(j)}} \right) = \begin{cases} 2 d_v 10^{\theta_v} log(10) \mathbf{\Psi}^{(i,j)} \left[1 - 10^{\theta_k} d_k^2 \right] & \text{if } v = k \\ 2 d_v 10^{\theta_v} log(10) \mathbf{\Psi}^{(i,j)} \left[-10^{\theta_k} d_k^2 \right] & \text{if } v \neq k \end{cases} \tag{31}$$

$$\frac{\partial}{\partial \theta_k} \left(\frac{\partial^2 \mathbf{\Psi}^{(i,j)}}{\partial x_u^{(i)} \partial x_v^{(j)}} \right) = \begin{cases} -4 d_u d_v 10^{\theta_u} 10^{\theta_v} log(10) \mathbf{\Psi}^{(i,j)} \left[1 - 10^{\theta_k} d_k^2 \right] & \text{if } u | v = k \\ 4 d_u d_v d_k^2 10^{\theta_u} 10^{\theta_v} 10^{\theta_k} log(10) \mathbf{\Psi}^{(i,j)} & otherwise \end{cases} \tag{32}$$

$$\frac{\partial}{\partial \theta_k} \left(\frac{\partial^2 \mathbf{\Psi}^{(i,j)}}{\partial x_{u=v}^{(i)} \partial x_{u=v}^{(j)}} \right) = \begin{cases} log(10) \mathbf{\Psi}^{(i,j)} \left[2(10^{\theta}) + 4(10^{3\theta}) d^4 - 10(10^{2\theta}) d^2 \right] & \text{if } (u = v) = k \\ - log(10) \mathbf{\Psi}^{(i,j)} 10^{\theta_k} d_k^2 \left[2(10^{\theta}) - 4(10^{2\theta}) d^2 \right] & \text{if } (u = v) \neq k \end{cases} \tag{33}$$

A.2 Matérn $\frac{5}{2}$ correlation function:

Gradient of correlation function with respect to X (i.e., cross-correlation):

$$\frac{\partial \mathbf{\Psi}^{(i,j)}}{\partial x^{(j)}} = \frac{5\theta d(\sqrt{5}a + 1) exp\left(-\sqrt{5}a\right)}{3} \tag{34}$$

Hessian of correlation function with respect to X (i.e., cross-correlation):

$$\frac{\partial^2 \mathbf{\Psi}^{(i,j)}}{\partial x_u^{(i)} \partial x_v^{(j)}} = \begin{cases} \frac{-25\theta_u \theta_v d_u d_v exp(-\sqrt{5}a)}{3} & \text{if } u \neq v \\ \left[\frac{-25\theta^2 d^2 + 5\theta(\sqrt{5}a + 1)}{3} \right] exp\left(-\sqrt{5}a\right) & \text{if } u = v \end{cases} \tag{35}$$

Derivative of correlation function with respect to θ_k:

$$\frac{\partial}{\partial \theta_k} \left(\psi_{\nu=5/2}(d_{u(v)}) \right) = \frac{-\left(5 + 5\sqrt{5}a\right) 10^{\theta} log(10) d_{u(v)}^2 exp\left(-\sqrt{5}a\right)}{6} \tag{36}$$

Derivatives of cross-correlation functions with respect to θ_k:

$$\frac{\partial}{\partial \theta_k} \left(\frac{\partial \mathbf{\Psi}^{(i,j)}}{\partial x_v^{(j)}} \right) = \begin{cases} 10^{\theta_v} d_v C_2 \left[C_1 + \left(\frac{-25}{6} \right) 10^{\theta_k} d_k^2 \right] & \text{if } v = k \\ 10^{\theta_v} 10^{\theta_k} d_v d_k^2 \left(\frac{-25 C_2}{6} \right) & \text{if } v \neq k \end{cases} \tag{37}$$

$$\frac{\partial}{\partial \theta_k} \left(\frac{\partial^2 \mathbf{\Psi}^{(i,j)}}{\partial x_u^{(i)} \partial x_v^{(j)}} \right) = \begin{cases} \frac{-25 C_2 \left(1 - \frac{\sqrt{5} 10^{\theta_k} d_k^2}{2a} \right) 10^{\theta_u} 10^{\theta_v} d_u d_v}{3} & \text{if } u | v = k \\ \frac{C_2 25\sqrt{5} 10^{\theta_u} 10^{\theta_v} 10^{\theta_k} d_u d_v d_k^2}{6a} & otherwise \end{cases} \tag{38}$$

$$\frac{\partial}{\partial \theta_k} \left(\frac{\partial^2 \mathbf{\Psi}^{(i,j)}}{\partial x_{u=v}^{(i)} \partial x_{u=v}^{(j)}} \right) = \begin{cases} V_3 + V_4 & \text{if } (u = v) = k \\ V_3 & \text{if } (u = v) \neq k \end{cases} \tag{39}$$

where

$$V_3 = \left[\left(\frac{25\sqrt{5}}{6a}\right)(10^{\theta_{u=v}})^2(d_{u=v})^2 - \left(\frac{25}{6}\right)10^{\theta_{u=v}}\right] C_2 10^{\theta_k} d_k^2 \tag{40}$$

$$V_4 = \left(\frac{-50C_2(10^{\theta})^2 d^2}{3}\right) + C_1 C_2 10^{\theta} \qquad C_1 = \left(\frac{5\sqrt{5}}{3}a + \frac{5}{3}\right) \qquad C_2 = log(10)exp\left(-\sqrt{5}a\right) \tag{41}$$

B Comparison of MLE and Least Squares Estimation (LSE) of scaling parameter (ρ)

Table 7: Comparison of NRMSE on the validation data set for different ways of ρ estimation: 1D function with $n_e = 4$, $n_c = 11$ and $n_p = 500$; Peaks, inductance (L) and quality factor (Q) functions with $n_e = 9$, $n_c = 30$ and $n_p = 500$ and $|S_{11}|_{mean}$ with $n_e = 30$, $n_c = 60$ and $n_p = 50$. LSE(C) and LSE(L) correspond to Least Squares Estimation with constant and linear distribution of ρ, respectively.

			NRMSE	(GECoK)		
	Gaussian			Matérn $\frac{5}{2}$		
	MLE	LSE(C)	LSE(L)	MLE	LSE(C)	LSE(L)
(1D)	**5.925e-04**	2.996e+00	6.145e-01	**3.949e-04**	4.861e+00	7.855e-01
(2D)	**7.614e-02**	1.463e-01	1.547e-01	**4.450e-02**	1.375e-01	1.383e-01
(L)	1.080e-03	4.625e-04	**4.608e-04**	1.243e-03	1.120e-03	**1.117e-03**
(Q)	**2.476e-03**	9.073e-03	9.065e-03	2.509e-03	1.398se-03	**1.393e-03**
(5D)	**4.279e-02**	4.477e-02	4.774e-02	**3.176e-02**	3.569e-02	3.481e-02
			NRMSE	(CoKriging)		
(1D)	**3.782e-03**	2.885e+01	3.134e+01	**3.619e-02**	3.122e+01	2.907e+01
(2D)	**1.198e-01**	1.790e-01	1.754e-01	**1.613e-01**	1.758e-01	1.766e-01
(L)	3.240e-03	**3.187e-03**	2.869e-03	**3.128e-03**	3.293e-03	3.238e-03
(Q)	**2.886e-03**	4.201e-03	4.652e-03	**1.918e-03**	3.081e-03	3.296e-03
(5D)	2.090e-01	2.088e-01	**2.087e-01**	1.159e-01	**9.347e-02**	1.091e-01

Table 8: Comparison of R^2 on the validation data set for different ways of ρ estimation: 1D function with $n_e = 4$, $n_c = 11$ and $n_p = 500$; Peaks, inductance (L) and quality factor (Q) functions with $n_e = 9$, $n_c = 30$ and $n_p = 500$ and $|S_{11}|_{mean}$ with $n_e = 30$, $n_c = 60$ and $n_p = 50$. LSE(C) and LSE(L) correspond to Least Squares Estimation with constant and linear distribution of ρ, respectively.

			R^2	(GECoK)		
	Gaussian			Matérn $\frac{5}{2}$		
	MLE	LSE(C)	LSE(L)	MLE	LSE(C)	LSE(L)
(1D)	**9.999e-01**	8.406e-01	9.673e-01	**9.999e-01**	7.4139e-01	9.582e-01
(2D)	**9.472e-01**	8.049e-01	7.818e-01	**9.819e-01**	8.276e-01	8.257e-01
(L)	**9.999e-01**	9.999e-01	9.999e-01	**9.999e-01**	9.999e-01	9.999e-01
(Q)	**9.999e-01**	9.989e-01	9.989e-01	**9.999e-01**	9.999e-01	9.999e-01
(5D)	**9.580e-01**	9.540e-01	9.477e-01	**9.769e-01**	9.708e-01	9.722e-01
			R^2	(CoKriging)		
(1D)	**9.998e-01**	0	0	**9.981e-01**	0	0
(2D)	**8.692e-01**	7.080e-01	7.197e-01	**7.627e-01**	7.183e-01	7.156e-01
(L)	**9.998e-01**	9.998e-01	9.998e-01	**9.998e-01**	9.998e-01	9.998e-01
(Q)	**9.998e-01**	9.997e-01	9.997e-01	**9.999e-01**	9.999e-01	9.999e-01
(5D)	0	8.456e-04	**1.324e-03**	6.919e-01	**7.996e-01**	7.267e-01

Table 9: Comparison of RAAE on the validation data set for different ways of ρ estimation: 1D function with $n_e = 4$, $n_c = 11$ and $n_p = 500$; Peaks, inductance (L) and quality factor (Q) functions with $n_e = 9$, $n_c = 30$ and $n_p = 500$ and $|S_{11}|_{mean}$ with $n_e = 30$, $n_c = 60$ and $n_p = 50$. LSE(C) and LSE(L) correspond to Least Squares Estimation with constant and linear distribution of ρ, respectively.

			RAAE	(GECoK)		
	Gaussian			Matérn $\frac{5}{2}$		
	MLE	LSE(C)	LSE(L)	MLE	LSE(C)	LSE(L)
(1D)	**3.812e-03**	2.188e-01	1.019e-01	**1.634e-03**	2.818e-01	1.201e-01
(2D)	**1.4296e-01**	3.529e-01	3.967e-01	**1.035e-01**	3.491e-01	3.209e-01
(L)	2.088e-03	**1.0346e-03**	1.037e-03	3.466e-03	2.327e-03	**2.322e-03**
(Q)	**7.130e-03**	1.889e-02	1.887e-02	3.962e-03	3.652e-03	**3.641e-03**
(5D)	1.579e-01	**1.395e-01**	1.461e-01	**1.199e-01**	1.246e-01	1.214e-01
			RAAE	(CoKriging)		
(1D)	**7.184e-03**	1.117e+00	1.066e+00	**1.929e-02**	1.189e+00	1.014e+00
(2D)	**2.761e-01**	4.696e-01	4.336e-01	**3.841e-01**	5.040e-01	5.032e-01
(L)	9.793e-03	8.387e-03	**7.946e-03**	9.280e-03	**8.537e-03**	8.709e-03
(Q)	**8.093e-03**	8.799e-03	9.404e-03	**5.207e-03**	7.448e-03	7.379e-03
(5D)	**8.266e-01**	8.327e-01	8.323e-01	4.004e-01	**3.561e-01**	4.018e-01

Table 10: Comparison of RMAE on the validation data set for different ways of ρ estimation: 1D function with $n_e = 4$, $n_c = 11$ and $n_p = 500$; Peaks, inductance (L) and quality factor (Q) functions with $n_e = 9$, $n_c = 30$ and $n_p = 500$ and $|S_{11}|_{mean}$ with $n_e = 30$, $n_c = 60$ and $n_p = 50$. LSE(C) and LSE(L) correspond to Least Squares Estimation with constant and linear distribution of ρ, respectively.

			RMAE	(GECoK)		
	Gaussian			Matérn $\frac{5}{2}$		
	MLE	LSE(C)	LSE(L)	MLE	LSE(C)	LSE(L)
(1D)	**1.291e-02**	1.101e+00	4.981e-01	**2.172e-02**	1.360e+00	5.394e-01
(2D)	**7.517e-01**	7.599e-01	9.738e-01	**3.739e-01**	7.886e-01	9.874e-01
(L)	2.725e-02	1.542e-02	**1.514e-02**	**1.775e-02**	2.816e-02	2.763e-02
(Q)	**4.965e-02**	1.540e-01	1.539e-01	2.114e-02	2.094e-02	**2.082e-02**
(5D)	**5.854e-01**	8.901e-01	9.735e-01	**4.786e-01**	6.143e-01	6.229e-01
			RMAE	(CoKriging)		
(1D)	**5.873e-02**	2.293e+00	3.209e+00	**1.815e-01**	2.279e+00	3.149e+00
(2D)	1.073e+00	**8.763e-01**	1.129e+00	1.064e+00	**8.466e-01**	9.295e-01
(L)	6.677e-02	6.191e-02	**5.000e-02**	5.480e-02	5.668e-02	**5.091e-02**
(Q)	**5.199e-02**	9.128e-02	1.024e-01	**2.702e-02**	5.050e-02	5.855e-02
(5D)	2.566e+00	**2.454e+00**	2.454e+00	1.385e+00	**1.219e+00**	1.338e+00