Struct Multidisc Optim (2012) 46:223–238
DOI 10.1007/s00158-012-0764-x

RESEARCH PAPER

An adaptive hybrid surrogate model

Jie Zhang · Souma Chowdhury · Achille Messac

Received: 15 March 2011 / Revised: 19 December 2011 / Accepted: 5 January 2012 / Published online: 31 January 2012

Abstract The determination of complex underlying relationships between system parameters from simulated and/or recorded data requires advanced interpolating functions, also known as surrogates. The development of surrogates for such complex relationships often requires the modeling of high dimensional and non-smooth functions using limited information. To this end, the hybrid surrogate modeling paradigm, where different surrogate models are combined, offers an effective solution. In this paper, we develop a new high fidelity surrogate modeling technique that we call the Adaptive Hybrid Functions (AHF). The AHF formulates a reliable Crowding Distance-Based Trust Region (CD-TR), and adaptively combines the favorable characteristics of different surrogate models. The weight of each contributing surrogate model is determined based on the local *measure of accuracy* for that surrogate model in the pertinent trust region. Such an approach is intended to exploit the advantages of each component surrogate. This approach seeks to simultaneously capture the global trend of the function as well as the local deviations. In this paper, the AHF combines four component surrogate models: (i) the Quadratic Response Surface Model (QRSM), (ii) the Radial Basis Functions (RBF), (iii) the Extended Radial Basis Functions (E-RBF), and (iv) the Kriging model. The AHF is applied to standard test problems and to a complex engineering design problem. Subsequent evaluations of the Root Mean Squared Error (RMSE) and the Maximum Absolute Error (MAE) illustrate the promising potential of this hybrid surrogate modeling approach.

Keywords Crowding distance · Hybrid surrogate modeling · Kriging · Radial basis functions · Response surface · Wind farm

1 Introduction

The need to quantify economic and engineering performance of complex systems often demands highly complex and computationally expensive simulations and/or expensive experiments.

Among the various approaches to deal with this problem, surrogate models have gained wide acceptance from the design community. Surrogate modeling is concerned with the construction of approximation models to estimate system performance, and to develop relationships between specific system inputs and outputs. Over the past two decades, function estimation methods and approximation-based optimization have progressed remarkably. Surrogate models are being extensively used in the analysis and optimization of computationally expensive simulation-based models. Surrogate modeling techniques have been used for a variety of applications from multidisciplinary design optimization to the reduction of analysis time and to the improvement of the tractability of complex analysis codes.

The general surrogate modeling problem can be stated as follows: "Given a set of data points $x^i \in R^m, i = 1, \cdots, n_p$, and the corresponding function values, $f(x^i)$, obtain a global approximation function, $\tilde{f}(x)$, that adequately represents the original function over a given design domain."

J. Zhang · S. Chowdhury
Department of Mechanical, Aerospace and Nuclear Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180, USA

A. Messac (✉)
Department of Mechanical and Aerospace Engineering, Syracuse University, Syracuse, NY 13244, USA
e-mail: messac@syr.edu

Several generalized function estimation techniques have been developed over the past few decades-e.g. Kriging, polynomial response surfaces, and radial basis functions. Advanced versions of these function estimation methods have been frequently reported in the literature, particularly those pertaining to the Multidisciplinary Design Optimization (MDO) community; in this community, function estimation is more popularly known as surrogate modeling. In the MDO field, the development of surrogate modeling is driven by the need to avoid the high computational expenses of employing complex system models. The new surrogate modeling approach developed in this paper is applicable to both system design and data mining problems.

The current paper presents a general approach for developing an effective surrogate model that seeks to take advantage of the positive characteristics of the well known surrogate modeling techniques. The remainder of the paper is organized as follows: Section 2 reviews existing surrogate modeling methods; Section 3 presents the motivation and the objectives of this research; the Adaptive Hybrid Functions (AHF) is formulated in Section 4; numerical experiments and results are presented in Sections 5 and 6.

2 Surrogate modeling review

A wide variety of surrogate modeling techniques have been reported in the literature, which include: (i) Polynomial Response Surface Model (PRSM) (Myers and Montgomery 2002), (ii) Kriging (Giunta and Watson 1998; Sakata et al. 2003; Simpson 1998; Cressie 1993), (iii) Radial Basis Functions (RBF) (Hardy 1971; Jin et al. 2001; Cherrie et al. 2002; Hussain et al. 2002), (iv) Extended Radial Basis Functions (E-RBF) (Mullur and Messac 2005, 2006; Zhang et al. 2010a, b, c), (v) Artificial Neural Networks (ANN) (Duda et al. 2000; Yegnanarayana 2004), and (vi) Support Vector Regression (SVR) (Clarke et al. 2005; Vapnik 1995; Duda et al. 2000; Basudhar and Missoum 2008). In the literature, the accuracy and effectiveness of various surrogate models for linear, nonlinear, smooth, and noisy responses have also been investigated (Forrester and Keane 2009; Queipo et al. 2005; Wang and Shan 2007; Simpson et al. 2008).

PRSM is a statistical tool that was primarily developed for fitting analytical models (typically quadratic polynomials) to an available data set. The classical PRSM is still one of the most widely used forms of surrogate models in engineering design (Forrester and Keane 2009; Queipo et al. 2005). PRSM is more suitable to capture the global trend and generally involves a small set of parameters (unknown coefficients). However, PRSM is often not adequate for capturing local accuracy in the close neighborhood of the training points. The challenge in accomplishing an exact fit has inspired researchers to explore the so-called kernel-based surrogate modeling techniques, which can provide an interpolating surface through the entire training data set. Kernel-based surrogate modeling techniques offer important advantages over the traditional PRSM such as: the ease of extending the estimated function to higher dimensions, and the representation of highly nonlinear functional relationships. Kernel-based surrogate modeling methods typically make use of local information related to each training data point, and combine this information to define the overall surrogate model. Kriging, RBF and E-RBF are popular kernel-based surrogate modeling techniques (Mullur and Messac 2005).

In surrogate-based modeling of complex problems, one of the common practices is (i) to construct one or more surrogate models, and (ii) to select the one with the best performance. This deterministic approach falls short of fully exploiting the resources available for surrogate modeling.

More recently, researchers have presented the development of a combination of different approximate models into a single hybrid model for developing weighted average surrogates (Zerpa et al. 2005; Goel et al. 2007; Sanchez et al. 2008; Acar and Rais-Rohani 2009; Viana et al. 2009; Acar 2010). Zerpa et al. (2005) showed one application using an ensemble of surrogate models to construct a weighted average surrogate for the optimization of alkaline-surfactant-polymer flooding processes. They found that the weighted average surrogate provides better performance than individual surrogates. Goel et al. (2007) considered an ensemble of three surrogate models (polynomial response surface, Kriging, and radial basis neural network), and used the Generalized Mean Square Cross-validation Error of individual surrogate models to select appropriate weight factors. Acar and Rais-Rohani (2009) treated the selection of weight factors in the general weighted-sum formulation of an ensemble as an optimization problem with the objective to minimize an error metric. The results showed that the optimized ensemble provides more accurate predictions than the stand-alone surrogate models. Acar (2010) investigated the efficiency of using various local error measures for constructing an ensemble of surrogate models, and also presented the use of the pointwise cross-validation error as a local error measure. Zhou et al. (2011) used a recursive process to obtain the values of the weights. The values of surrogate weights are updated in each iteration until the last ensemble achieves a desirable prediction accuracy.

3 Research objectives

Building on previous research, this paper offers an innovative approach to develop an effective hybrid surrogate model, the Adaptive Hybrid Functions (AHF).

Each surrogate modeling methodology (or underlying basis function) has its own benefits and limitations; hence, these methods are expected to present widely different levels of numerical fidelity for different types of problems. We hypothesized that, an approach that adaptively combines the favorable characteristics of characteristically different surrogate modeling methods (a hybrid surrogate) would be able to address a broad range of applications (that demand function estimation). Assuming that the user does not have definitive insight into the functional relationships that we are seeking to model, the measured (or simulated) sample data is all the information that we have at our disposal. With that understanding, the broad (and often contradicting) objectives of surrogate modeling can be viewed as comprising two parts:

1. To accomplish reasonable local accuracy in the neighborhood of the training points, and
2. To capture the global trend of the functional variation.

Simultaneous fulfillment of the above objectives presents significant challenges to surrogate modeling approaches. In the context of these challenges, improving the overall fidelity of the estimated function, at the same computational expense as that of the individual surrogates, would be particularly helpful. To this end, a hybrid surrogate model provides an effective solution.

The hybrid surrogate model presented in this paper makes the following key contributions:

1. An adaptive hybrid surrogate model is developed.
2. Both the global and the local accuracies are captured by employing the density of training points.
3. Crowding distance-based trust region boundaries are formulated.
4. The weights of component surrogates are adaptively selected.
5. Local accuracy with function continuity is retained.
6. The model is also suitable for problems with limited control over design of experiments.

4 Adaptive hybrid functions (AHF)

It is important to first comment on the difference between the global and the local accuracies of a surrogate, since they are motivationally and philosophically distinct. Global accuracy indicates that the surrogate can capture the global trend for a given set of simulated or experimental data; errors between the estimated and the actual functions may exist at the training points in this case, e.g. PRSM and SVR. Local accuracy generally reflects that the surrogate has zero errors at all training points and a likely higher accuracy around the training points, e.g. Kriging, RBF and E-RBF. The AHF methodology developed in this paper seeks to simultaneously capture the global and the local accuracies. The AHF surrogate combines (i) Quadratic Response Surface Model (QRSM), (ii) Radial Basis Functions (RBFs), (iii) Extended Radial Basis Functions (E-RBF), and (iv) Kriging.

This hybrid surrogate modeling methodology follows a three-step approach:

1. Determination of a trust region: numerical bounds of the output as a function of the input vector over the feasible input space.
2. Characterization of the *local measure of accuracy* (using kernel functions) of the estimated function value, and the representation of the corresponding kernel function parameters as functions of the input vector.
3. Generation of different surrogate models (component surrogates), and weighted aggregation of the estimated function value based on the *local measure of accuracy* of the individual surrogates.

The primary objective of this paper is to lay the foundation for an adaptive hybrid surrogate model. In addition, we also develop novel strategies to implement the three-step approach.

We will consider a set of training points, D, expressed as

$$D = \begin{pmatrix} x_1^1 & x_2^1 & \cdots & x_{n_d}^1 & y^1 \\ x_1^2 & x_2^2 & \cdots & x_{n_d}^2 & y^2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_1^{n_p} & x_2^{n_p} & \cdots & x_{n_d}^{n_p} & y^{n_p} \end{pmatrix}$$

The three steps, followed to formulate the AHF surrogate model, are illustrated in Fig. 1. In the representation given by D, x_j^i is the jth dimension of the input vector that represents the ith training point, and y^i is the corresponding output; n_d represents the dimension of the input variable, and n_p represents the number of training data points.

For the sake of simplicity of presentation, we use a two-dimensional scenario to illustrate the steps of the AHF formulation. The function used for this purpose is test function 1; details of this function are provided in Section 5.

4.1 Step A.1: Determination of the base model

The base model is developed using smooth functions to obtain a global approximation for the given set of points, D. This base model is intended to capture the global trend of the training points, thereby addressing the global accuracy for the overall surrogate. In the current paper, the base model (solid line in Fig. 2-in 2D scenario) is constructed using QRSM. However, the base model also has the flexibility

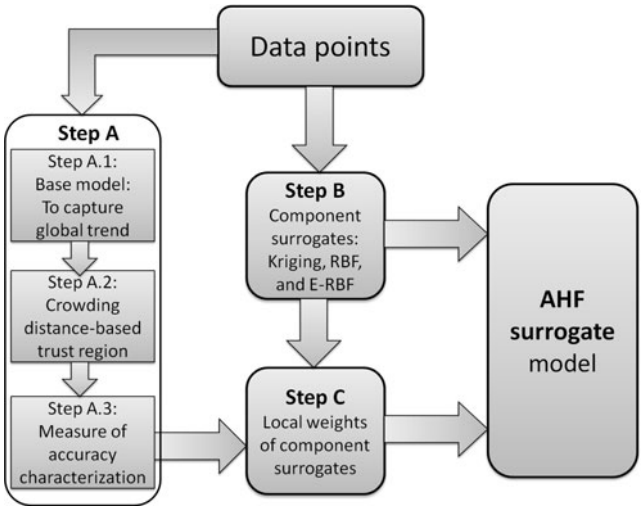


Fig. 1 The framework of the AHF surrogate model

to use other smooth functions. A typical QRSM can be represented as

$$\tilde{f}_{\text{qrsm}}(x) = a_0 + \sum_{i=1}^{n_d} b_i x_i + \sum_{i=1}^{n_d} c_{ii} x_i^2 + 2 \sum_{i=1}^{n_d-1} \sum_{j>i}^{n_d} c_{ij} x_i x_j \tag{1}$$

where the x_i's are the input parameters; and the generic variables a_0, b_i, c_{ii} and c_{ij} are the unknown coefficients determined by the least squares approach.

4.2 Step A.2: Formulation of trust region boundaries

In this step, we formulate a reliable trust region for surrogate modeling. The trust region, which we call the Crowding Distance-based Trust Region (CD-TR), is a novel contribution of this paper. The boundaries of the trust region

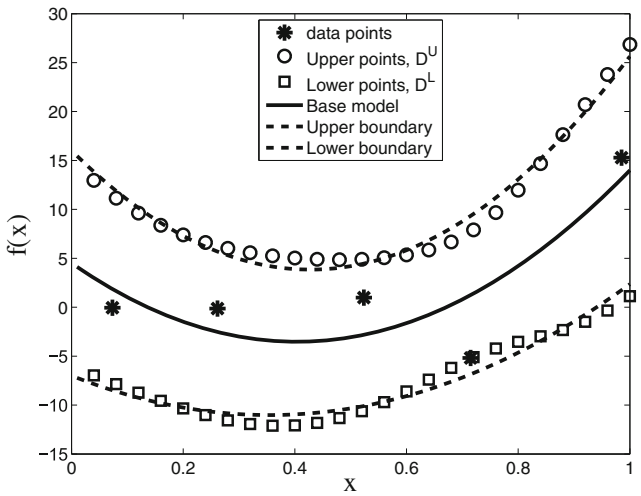


Fig. 2 Base and boundary models of AHF surrogate

are adaptively constructed based on the base model. This approach seeks to significantly reduce the "deviation of the final surrogate from the training data". The final hybrid surrogate model adaptively aggregates three nonparametric surrogate modeling techniques (Kriging, RBF, E-RBF) within the bounds of the CD-TR.

In this paper, the boundaries of the CD-TR are constructed according to the base model and the crowding distance evaluation. The crowding distance of a point represents the density of points around that point. A set of points are selected on the base model, and the crowding distance is evaluated for each point. Then, the base model is relaxed along the positive and negative directions of the output axis, to obtain the boundaries of the surrogate (dashed lines in Fig. 2).

In the Non-dominated Sorting Genetic Algorithm (NSGA-II), the crowding distance value of a candidate solution provides a local estimate of the density of solutions (Deb 2001). In this paper, crowding distance is used to evaluate the density of training points surrounding any point on the base model (the solid line in Fig. 2).

1. A large crowding distance value of a point reflects low sample density (fewer points around that point), and the *measure of accuracy* of the surrogate is expected to be relatively lower around that point. Therefore, we need wider boundaries at that point.
2. A small crowding distance value of a point reflects high sample density (more points around that point), and the *measure of accuracy* of the surrogate is expected to be relatively higher around that point. Therefore, tighter boundaries can be employed.

Based on the crowding distance value of each point on the base model, we construct adaptive boundaries of the CD-TR. In this paper, the crowding distance of the ith point on the base model (CD^i) is evaluated in the trust region by

$$CD^i = \sum_{j=1}^{n_p} \left\| x^j - x^i \right\|^2 \tag{2}$$

where n_p is the number of training data points. A parameter ρ is defined to represent the local density of input data, as given by

$$\rho^i = \frac{1}{CD^i} \tag{3}$$

The parameter ρ is then normalized to obtain distance coefficients α_i's, as given by

$$\alpha_i = \frac{\max(\rho) - \rho^i}{\max(\rho) - \min(\rho)} \tag{4}$$

The adaptive distance d^i between the i^{th} corresponding point on the boundary and the base model along the positive or negative direction of the output axis, is expressed as

$$d^i = (1 + \alpha_i) \times \max_{j \in D} \left| \tilde{f}_{\text{qrsm}}(x^j) - y^j \right| \tag{5}$$

where D represents the original training data set. It is helpful to note that, in (5), the index j represents training points and the index i represents a uniform set of points selected on the base model. In (5), the adaptive distance is divided into two parts:

1. $\max_{j \in D} \left| \tilde{f}_{\text{qrsm}}(x^j) - y^j \right|$, a scaling constant that ensures that all the training points are located between the boundaries; and
2. $\alpha_i \times \max_{j \in D} \left| \tilde{f}_{\text{qrsm}}(x^j) - y^j \right|$, an adaptive distance based on the distance coefficients (α_i's).

Crowding distance is evaluated with respect to each of the selected points, based on which α is formulated. The extent of the boundary region is scaled using the maximum of the training data deviation from the base model. Here, $\tilde{f}_{\text{qrsm}}(x^j)$ is the estimated output value of the jth training point using QRSM; $\left| \tilde{f}_{\text{qrsm}}(x^j) - y^j \right|$ is the distance from the jth training point to the base model along the direction of the output axis. Subsequently, we obtain two sets of points, D^U and D^L, for constructing the two boundaries, expressed as

$$D^U = \begin{pmatrix} x_1^1 & x_2^1 & \cdots & x_{n_d}^1 & y^1 + d^1 \\ x_1^2 & x_2^2 & \cdots & x_{n_d}^2 & y^2 + d^2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_1^{n_p} & x_2^{n_p} & \cdots & x_{n_d}^{n_p} & y^{n_p} + d^{n_p} \end{pmatrix}$$

$$D^L = \begin{pmatrix} x_1^1 & x_2^1 & \cdots & x_{n_d}^1 & y^1 - d^1 \\ x_1^2 & x_2^2 & \cdots & x_{n_d}^2 & y^2 - d^2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_1^{n_p} & x_2^{n_p} & \cdots & x_{n_d}^{n_p} & y^{n_p} - d^{n_p} \end{pmatrix}$$

We recall that QRSM is adopted to estimate the upper boundary surface (upper dashed line in Fig. 2) using the generated data points D^U, and the lower boundary surface (lower dashed line in Fig. 2) using the generated data points D^L. Within the trust region boundaries, we estimate the probabilities associated with individual component surrogates in the following step.

It is important to note that the CD-TR estimation is particularly useful for recorded or measured data-based (commercial or experimental data) surrogate modeling. In the case of problems where the user has control over sampling, the initial sample data is expected to be relatively evenly distributed, and significant variation in crowding distance may not be observed.

4.3 Step A.3: Estimation of measure of accuracy

With the CD-TR developed above, it is important to determine the *measure of accuracy* of the estimated function value at a given point in the trust region. Based on the *measure of accuracy*, we can adaptively combine different component surrogate models. In this paper, we develop a metric, which we call the Accuracy Measure of Surrogate Modeling (AMSM), to represent the uncertainty in the estimated function value.

Function estimation is performed between the two boundary surfaces, using a "*local measure of accuracy*" technique. The uncertainty in the estimated function value at a location in the input variable domain is modeled using a kernel function. This kernel function is expressed as a function of the output parameter. The corresponding coefficients of the kernel function are represented as functions of the input vector, thereby characterizing the *measure of accuracy* of the estimated function over the entire input domain.

The kernel function used to represent the measure of accuracy must have the following properties.

1. The kernel function value must be a maximum of one at the actual output value, $y(x^i)$.
2. The kernel function must be equal to the specified small tolerance value at the upper and the lower boundaries of the trust region.
3. The function must increase monotonically from either boundary to the actual output value.
4. The function must be continuous within the CD-TR bounds.

An example of the kernel function for a 2D problem (test function 1) is shown in Fig. 3. In this paper, the following kernel function is adopted. This kernel function, which satisfies the specified requirements, is expressed as

$$P(z) = a \exp\left[-\frac{(z - \mu)^2}{2\sigma^2} \right] \tag{6}$$

where the amplitude coefficient a is set to one; the coefficients μ and σ represent the mean and the standard deviation of the kernel function, respectively. It is helpful to note that other kernel functions that have similar properties can also be used to represent the *measure of accuracy*.

The distance between the two boundaries is normalized. At each training data point x^i, the output value at the lower and upper boundaries ($f_L^i(x^i)$ and $f_U^i(x^i)$, respectively) are also normalized. We assume that the estimated *measure of*

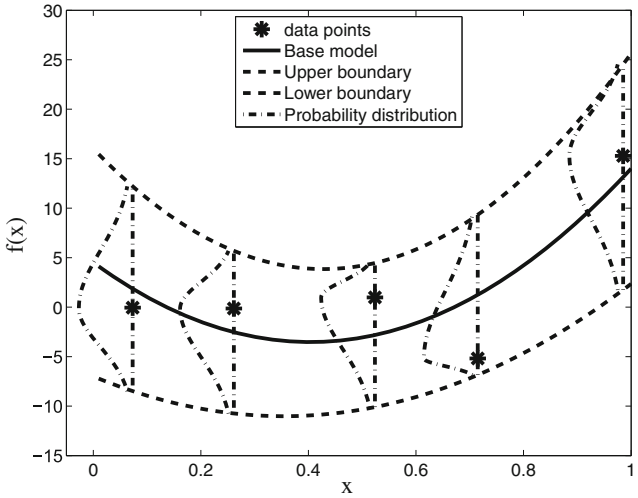


Fig. 3 Measure of accuracy evaluation of the AHF surrogate

accuracy (kernel function) has a maximum of one at the actual output value $y(x^i)$; and, a minimum of 0.1 at the boundaries (within the trust region). The output value of a training point does not necessarily occur midway between the two boundaries. In order to ensure the continuity of the kernel function, we divide the function into two parts, with distinct standard deviations and the same mean. We can therefore represent the kernel function as

$$P(x^i) = \begin{cases} a \exp\left\{-\frac{[y(x^i)-\mu(x^i)]^2}{2\sigma_1^2(x^i)}\right\} & \text{if } 0 \le y(x^i) \le \mu(x^i) \\ a \exp\left\{-\frac{[y(x^i)-\mu(x^i)]^2}{2\sigma_2^2(x^i)}\right\} & \text{if } \mu(x^i) \le y(x^i) \le 1 \end{cases} \tag{7}$$

where the parameters σ_1 and σ_2, controlled by the full width at one tenth maximum (Δx_{10}) requirement, are given by

$$\sigma_1(x^i) = \frac{\Delta z_{10}(x^i)}{2\sqrt{2\ln 10}} = \frac{2\left[\mu(x^i) - f_L^i(x^i)\right]}{2\sqrt{2\ln 10}} = \frac{2\mu(x^i)}{2\sqrt{2\ln 10}} = \frac{\mu(x^i)}{\sqrt{2\ln 10}}, \quad \text{and} \tag{8}$$

$$\sigma_2(x^i) = \frac{\Delta z_{10}(x^i)}{2\sqrt{2\ln 10}} = \frac{2\left[f_U^i(x^i) - \mu(x^i)\right]}{2\sqrt{2\ln 10}} = \frac{2\left[1-\mu(x^i)\right]}{2\sqrt{2\ln 10}} = \frac{1-\mu(x^i)}{\sqrt{2\ln 10}}, \quad \text{where} \tag{9}$$

$$P(\mu \pm 0.5\Delta z_{10}) = \frac{1}{10} \tag{10}$$

From (7), we can determine the *measure of accuracy* coefficients $\mu(x^i)$ for the ith training point. The coefficient μ is then expressed in terms of input variables x_j^i using a Polynomial Response Surface.

A trust region for surrogate modeling can be determined using the CD-TR technique; and the *measure of accuracy* can be estimated using the AMSM method. This accuracy modeling approach is somewhat similar to the variance estimation in Kriging, introduced in the Efficient Global Optimization (EGO) approach (Jones et al. 1998). In the stochastic process approach, the authors assume that the correlation between errors is related to the distance between the corresponding points. Special weighted distance formula, rather than Euclidean distance, is used to account for the weights of different variables. Since then, the EGO approach has become a widely used technique in engineering applications (Huang et al. 2006; Bichon et al. 2008, 2009; Viana et al. 2010).

4.4 Step B: Creation of surrogate models using existing methods

In this step, we construct different surrogate models (component surrogates). The selected component surrogate models are expected to be locally accurate - with greater accuracy in the local region close to the training points. Three component surrogates are constructed based on the set of training points D, using Kriging, RBF, and E-RBF. However, we can also combine other standard surrogates that are locally accurate. For each test point, the estimated function vector can be represented as $\tilde{f} = \left\{\tilde{f}_{\text{kriging}}\ \tilde{f}_{\text{rbf}}\ \tilde{f}_{\text{erbf}}\right\}$. The parameters $\tilde{f}_{\text{kriging}}$, $\tilde{f}_{\text{rbf}}$ and $\tilde{f}_{\text{erbf}}$ represent function values estimated by the Kriging, the RBF and the E-RBF methods, respectively.

The three component surrogates can offer different levels of numerical fidelity. In general, Kriging models are more accurate for nonlinear problems (Wang and Shan 2007). The RBF approach can accurately model scattered multivariate data. The RBF method is also relatively easier to construct compared to Kriging (Queipo et al. 2005; Mullur and Messac 2005). The E-RBF approach provides additional degrees of freedom on an RBF surrogate to ensure that certain desirable properties, such as nonlinearity and convexity, are addressed.

4.4.1 Kriging

Kriging (Giunta and Watson 1998; Sakata et al. 2003; Simpson 1998) is an approach to approximate irregular data. The kriging approximation function consists of two components: (i) a global trend function, and (ii) a deviation function representing the departure from the trend function.

The trend function is generally a polynomial (e.g., constant, linear, or quadratic). This approach is particularly useful for predicting temporally and spatially correlated data (Simpson 1998). The general form of the kriging surrogate model is given by (Cressie 1993):

$$\tilde{f}(x) = G(x) + Z(x) \tag{11}$$

where $\tilde{f}(x)$ is the unknown function of interest, $G(x)$ is the known approximation (usually polynomial) function, and $Z(x)$ is the realization of a stochastic process with a zero mean, and a nonzero covariance. The i, j-th element of the covariance matrix of $Z(x)$ is given by

$$COV[Z(x^i), Z(x^j)] = \sigma_z^2 R_{ij} \tag{12}$$

where R_{ij} is the correlation function between the ith and the jth data points; and σ_z^2 is the process variance. In the present paper, a Gaussian function is used as the correlation function, defined by

$$R(x^i, x^j) = R_{ij} = \exp\left\{-\Sigma_{k=1}^{n_d}\theta_k\left(x_k^i - x_k^j\right)^2\right\} \tag{13}$$

where θ_k is distinct for each dimension, and these unknown parameters are generally obtained by solving a nonlinear optimization problem.

4.4.2 Radial basis functions

The idea of using Radial Basis Functions (RBF) as approximation functions was introduced by Hardy (1971), where he used the multiquadric RBF to fit irregular topographical data. Since then, RBF has been used for various applications that require global approximations of multidimensional scattered data (Jin et al. 2001; Cherrie et al. 2002; Hussain et al. 2002).

RBF is expressed in terms of the Euclidean distance, $r = \|x - x^i\|$, of a point x from a given data point, x^i. One of the most effective forms is the multiquadric function (Cherrie et al. 2002; Hardy 1971), which is defined as

$$\psi(r) = \sqrt{r^2 + c^2} \tag{14}$$

where $c > 0$ is a prescribed real valued parameter. The final approximation function is a linear combination of these basis functions across all data points, as given by

$$\tilde{f}(x) = \sum_{i=1}^{n_p} \sigma_i \psi\left(\|x - x^i\|\right) \tag{15}$$

where σ_i's are the unknown coefficients (to be determined), and n_p denotes the number of selected data points. In this case, the number of coefficients is equal to the number of sample points, n_p. Equation (15) can be solved using the pseudo inverse method.

4.4.3 Extended radial basis functions

The Extended Radial Basis Functions (E-RBF) (Mullur and Messac 2005) approach uses a combination of the radial and the non-radial basis functions. The Non-Radial Basis Functions (N-RBF) are not functions of the Euclidean distance, r. Instead, they are functions of individual coordinates of generic points x relative to a given data point x^i, in each dimension separately. We define the coordinate vector as $\xi^i = x - x^i$, which is a vector of n_d elements, each corresponding to a single coordinate dimension. Thus, ξ_j^i is the coordinate of any point x relative to the data point x^i along the jth dimension. The N-RBF for the ith data point and the jth dimension is denoted by ϕ_{ij}. It is composed of three distinct components, as given by

$$\phi_{ij}(\xi_j^i) = \alpha_{ij}^L \phi^L(\xi_j^i) + \alpha_{ij}^R \phi^R(\xi_j^i) + \beta_{ij}\phi^\beta(\xi_j^i) \tag{16}$$

where α_{ij}^L, α_{ij}^R and β_{ij} are coefficients to be determined for the given problem. The parameters ϕ^L, ϕ^R and ϕ^β are defined in Table 1.

The E-RBF approach presents a linear combination of the RBF and the N-RBF. The approximation function takes the form

$$\tilde{f}(x) = \sum_{i=1}^{n_p} \sigma_i \psi\left(\|x - x^i\|\right) + \sum_{i=1}^{n_p}\sum_{j=1}^{n_d}\left\{\alpha_{ij}^L \phi^L(\xi_j^i) + \alpha_{ij}^R \phi^R(\xi_j^i) + \beta_{ij}\phi^\beta(\xi_j^i)\right\} \tag{17}$$

where ϕ^L, ϕ^R and ϕ^β are components of the N-RBF. The vectors α^L, α^R and β, defined above, contain $n_d n_p$ elements each, and the vector σ contains n_p coefficients. Thus,

Table 1 Non-Radial basis functions

Region	Range of ξ_j^i	ϕ^L	ϕ^R	ϕ^β
I	$\xi_j^i \le -\lambda$	$(-t\lambda^{t-1})\xi_j^i + \lambda^t(1-t)$	0	ξ_j^i
II	$-\lambda \le \xi_j^i \le 0$	$(\xi_j^i)^t$	0	ξ_j^i
III	$0 \le \xi_j^i \le \lambda$	0	$(\xi_j^i)^t$	ξ_j^i
IV	$\xi_j^i \ge \lambda$	0	$(t\lambda^{t-1})\xi_j^i + \lambda^t(1-t)$	ξ_j^i

λ, t: Prescribed parameters

the total number of coefficients to be determined is equal to $(3n_d+1)n_p$. Two methods that can be used to solve (17) are (i) linear programming, and (ii) pseudo inverse. Mullur and Messac (2005) provides the details of the E-RBF approach.

4.5 Step C: Determining local weights of component surrogates

Finally, we formulate the Adaptive Hybrid Functions (AHF) surrogate model by adaptive selection of weights for the three component surrogate models (RBF, E-RBF and Kriging). The AHF is a weighted summation of function values estimated by the component surrogates, as given by

$$\tilde{f}_{\text{AHF}} = \sum_{i=1}^{n_s} w_i(x)\,\tilde{f}_i(x) \tag{18}$$

where n_s is the number of component surrogates combined into the AHF, and $\tilde{f}_i(x)$ represents the estimated value by each component surrogate. The weights w_i's are expressed in terms of the estimated *measure of accuracy*, which is given by

$$w_i(x) = \frac{P_i(x)}{\sum_{i=1}^{n_s} P_i(x)} \tag{19}$$

where $P_i(x)$ is the *measure of accuracy* of the ith surrogate for point x.

In this paper, three surrogates (RBF, E-RBF and Kriging) are combined into the AHF ($n_s = 3$), and the final AHF surrogate model (for test function 1) is shown in Fig. 4. Figure 5 illustrates the weight values (w_i) of each component surrogate model over the entire domain.

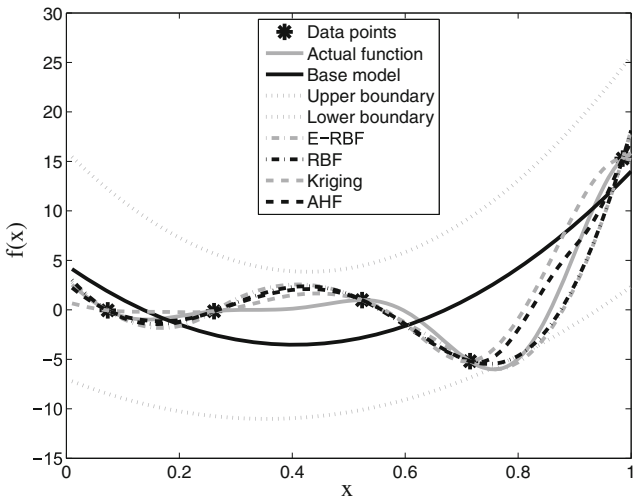


Fig. 4 Adaptive Hybrid Functions

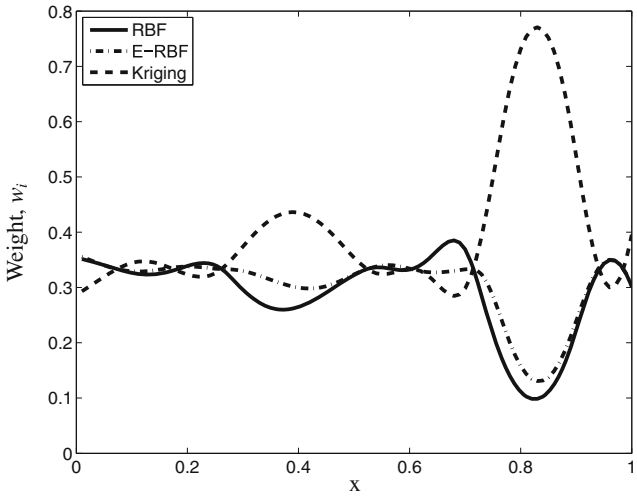


Fig. 5 The weight values of each component surrogate

5 Numerical examples

In this section, we compare the performance of the Adaptive Hybrid Functions (AHF) developed in this paper, with the stand-alone surrogate models, namely: (i) QRSM, (ii) RBF, (iii) E-RBF, and (iv) Kriging. Six benchmark problems and an engineering design problem are tested in this regard.

5.1 Benchmark problems

The performance of the AHF is illustrated using the following analytical benchmark problems: (i) 1-variable function (Forrester et al. 2008), (ii) 2-variable function (Mullur and Messac 2005), (iii) Goldstein & Price function (Goel et al. 2007), (iv) Branin–Hoo function (Goel et al. 2007), (v) Hartmann function with three variables (Goel et al. 2007), and (vi) Hartmann function with six variables (Goel et al. 2007). The expressions of these problems are summarized as follows:

Test Function 1: 1-Variable Function

$$f(x) = (6x_1 - 2)^2 \sin\left[2\,(6x_1 - 2)\right] \quad \text{where} \quad x_1 \in [0\ \ 1] \tag{20}$$

Test Function 2: 2-Variable Function

$$f(x) = [30 + x_1 \sin(x_1)] \times \left[4 + \exp\left(-x_2^2\right)\right] \quad \text{where} \quad x_1 \in [0\ \ 10],\quad x_2 \in [0\ \ 10] \tag{21}$$

Test Function 3: Goldstein & Price Function

$$f(x) = \Big[1 + (x_1 + x_2 + 1)^2 \times \left(19 - 14x_1 + 13x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2\right)\Big] \times \Big[30 + (2x_1 - 3x_2)^2 \times \left(18 - 32x_1 + 12x_1^2 - 48x_2 - 36x_1x_2 + 27x_2^2\right)\Big] \quad \text{where} \quad x_1 \in [-2\ \ 2], \quad x_2 \in [-2\ \ 2] \tag{22}$$

Test Function 4: Branin–Hoo Function

$$f(x) = \left(x_2 - \frac{5.1x_1^2}{4\pi^2} + \frac{5x_1}{\pi} - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10 \quad \text{where} \quad x_1 \in [-5\ \ 10], \quad x_2 \in [0\ \ 15] \tag{23}$$

Test Function 5 and 6: Hartmann Function

$$f(x) = -\sum_{i=1}^{4} c_i \ \exp\left\{-\sum_{j=1}^{n} A_{ij}\left(x_j - P_{ij}\right)^2\right\} \quad \text{where} \quad x = (x_1\ \ x_2\ \ \ldots\ \ x_n) \quad x_i \in [0\ \ 1] \tag{24}$$

Two instances of this problem are considered based on the number of design variables, (i) Hartmann-3 with three input variables (test function 5), and (ii) Hartmann-6 with six input variables (test function 6). When the number of variables, $n = 3$, c is given by $c = [1\ 1.2\ 3\ 3.2]^T$, and A and P are given by

$$A = \begin{bmatrix} 3.0 & 10 & 30 \\ 0.1 & 10 & 35 \\ 3.0 & 10 & 30 \\ 0.1 & 10 & 35 \end{bmatrix},$$

$$P = \begin{bmatrix} 0.3689 & 0.1170 & 0.2673 \\ 0.4699 & 0.4387 & 0.7470 \\ 0.1091 & 0.8732 & 0.5547 \\ 0.03815 & 0.5743 & 0.8828 \end{bmatrix}$$

When the number of variables, $n = 6$, c is given by $c = [1\ 1.2\ 3\ 3.2]^T$, and A and P are given by

$$A = \begin{bmatrix} 10.0 & 3.0 & 17.0 & 3.5 & 1.7 & 8.0 \\ 0.05 & 10.0 & 17.0 & 0.1 & 8.0 & 14.0 \\ 3.0 & 3.5 & 1.7 & 10.0 & 17.0 & 8.0 \\ 17.0 & 8.0 & 0.05 & 10.0 & 0.1 & 14.0 \end{bmatrix}$$

$$P = \begin{bmatrix} 0.1312 & 0.1696 & 0.5569 & 0.0124 & 0.8283 & 0.5886 \\ 0.2329 & 0.4135 & 0.8307 & 0.3736 & 0.1004 & 0.9991 \\ 0.2348 & 0.1451 & 0.3522 & 0.2883 & 0.3047 & 0.6650 \\ 0.4047 & 0.8828 & 0.8732 & 0.5743 & 0.1091 & 0.0381 \end{bmatrix}$$

5.2 Wind farm power generation model

In this paper, we use the AHF method to accurately represent the power generated by a wind farm as a function of the locations of the turbines in the farm (farm layout). The flow pattern inside a wind farm is complex, primarily due to the wake effects and the highly turbulent flow. The power generated by a wind farm (P_{farm}) comprised of N wind turbines is evaluated as a sum of the powers generated by the individual turbines, which is expressed as

$$P_{\text{farm}} = \sum_{j=1}^{N} P_j \tag{25}$$

Accordingly, the farm efficiency can be expressed as

$$\eta_{\text{farm}} = \frac{P_{\text{farm}}}{\sum_{j=1}^{N} P_{0j}} \tag{26}$$

where P_{0j} is the power that turbine-j would generate if operating as a stand-alone entity, for the given incoming wind velocity. Detailed formulation of the power generation model can be found in the papers by Chowdhury et al. (2010, 2012).

The power generated is thus a function of the location co-ordinates of each turbine. With the turbine-types and operating conditions remaining fixed, the x-y co-ordinates are the design variables for wind farm layout optimization. Therefore, we develop a hybrid response surface (using AHF) to represent the net power generation as a function of the turbine location co-ordinates. In the case of a wind farm comprising N turbines, the power generation model presents a $2N$ dimensional problem.

5.3 Sampling strategies

In the case of problems with simulated data, the choice of an appropriate sampling technique is generally considered crucial for the performance of any surrogate modeling approach.

5.3.1 Latin hypercube sampling

Latin hypercube sampling, is a strategy for generating random sample points, and promotes a uniform representation of the entire variable domain (McKay et al. 1979). A Latin hypercube sample containing n_p sample points (between 0 and 1) over m dimensions is a matrix of n_p rows and

m columns. Each row corresponds to a sample point. The values of n_p points in each column are randomly selected-one from each of the intervals, $(0, 1/n_p)$, $(1/n_p, 2/n_p)$, $\ldots$, $(1 - 1/n_p, 1)$ (McKay et al. 1979). In this paper, Latin hypercube sampling is used to generate the training and the test data for the benchmark problems.

5.3.2 Sobol's quasirandom sequence generator

The design variables of the wind farm power generation problem are sampled using the Sobol's quasirandom sequence generator (Sobol 1976). Sobol sequences use a base of two to form successively finer uniform partitions of the unit interval, and reorder the coordinates in each dimension. The algorithm for generating Sobol sequences is discussed in Bratley and Fox, Algorithm 659 (Bratley and Fox 1988).

5.4 Selection of parameters

Through numerical experiments, we found that the following prescribed coefficient values generally produced accurate function estimations. We set $c = 0.9$ for the RBF approach. We use $c = 0.9$ and $\lambda = 4.75$ for the E-RBF approach. The parameter t of the E-RBF approach is fixed at 2 (second degree monomial). The prescribed values are shown in Table 2.

For the Kriging method used in this paper, we use an efficient MATLAB implementation, DACE (design and analysis of computer experiments), developed by Lophaven et al. (2002). The bounds on the correlation parameters in the nonlinear optimization, θ_l and θ_u, are selected as 0.1 and 20, respectively. Under the Kriging approach, the order of the global polynomial trend function was specified to be zero.

5.5 Performance criteria

The overall performance of the surrogates is evaluated by applying the following two standard performance metrics to the test data points: (i) Root Mean Squared Error (RMSE) (Jin et al. 2001; Forrester and Keane 2009), which provides a global error measure over the entire design domain, and (ii) Maximum Absolute Error (MAE) (Queipo et al. 2005; Mullur and Messac 2006), which is indicative of local deviations. To compare the performances of different methods across functions, we normalize the RMSE measure and the MAE measure using the actual function values.

Table 2 Parameter selection for the E-RBF method

Parameter	λ	c	t
Value	4.75	0.9	2

5.5.1 Root Mean Squared Error (RMSE)

The RMSE is given by

$$RMSE = \sqrt{\frac{1}{n_t}\sum_{k=1}^{n_t}\left(f(x^k) - \tilde{f}(x^k)\right)^2} \tag{27}$$

where $f(x^k)$ represents the exact function value for the test point x^k, $\tilde{f}(x^k)$ is the corresponding estimated function value, and n_t is the number of test points chosen for evaluating the error measure. The Normalized Root Mean Squared Error (NRMSE) is given by

$$NRMSE = \sqrt{\frac{\sum_{k=1}^{n_t}\left(f(x^k) - \tilde{f}(x^k)\right)^2}{\sum_{k=1}^{n_t}\left(f(x^k)\right)^2}} \tag{28}$$

5.5.2 Maximum Absolute Error (MAE)

The MAE and the Normalized Maximum Absolute Error (NMAE) are expressed as

$$MAE = \max_k \left|f(x^k) - \tilde{f}(x^k)\right| \tag{29}$$

$$NMAE = \frac{\max_k \left|f(x^k) - \tilde{f}(x^k)\right|}{\sqrt{\frac{1}{n_t}\sum_{k=1}^{n_t}\left(f(x^k) - \bar{f}\right)^2}} \tag{30}$$

where $\bar{f}$ is the mean of the actual function values for the n_t test points.

5.6 Numerical settings

The numerical settings used to fit surrogate models for each problem are given in Table 3, which lists (i) the number of input variables, (ii) the number of training points, and (iii) the number of test points for each test problem.

For the wind farm layout optimization problem, we test the surrogates for a nine-turbine wind farm. Hence, the number of input variables is eighteen ($N_{var} = 18$). The wind velocity data is obtained from the North Dakota Agricultural Weather Network (NDAWN) (NDAWN 2010).

Table 3 Numerical setup for test problems

Function	No. of variables	No. of training points	No. of test points
Test function 1	1	5	100
Test function 2	2	36	441
Goldstein & Price function	2	36	1681
Branin–Hoo function	2	36	961
Hartmann-3 function	3	49	216
Hartmann-6 function	6	150	729
Wind farm problem	18	200	100

6 Results and discussion

6.1 Benchmark problems

Table 4 shows the RMSE and the NRMSE estimated by each surrogate model for the six benchmark test functions. The least RMSE and NRMSE values obtained for each function are shown in boldface in Table 4. The values of the MAE and the NMAE are listed in Table 5. The least MAE and NMAE values obtained are also shown in boldface in the table. The comparison of the performance of the individual surrogates and the AHF is illustrated through bar diagrams in Fig. 6.

The comparison results are also presented as relative differences comparing the AHF surrogate to other surrogates. The relative difference in NRMSE (e^{NRMSE}) of each surrogate is given by

$$e_i^{\text{NRMSE}} = \frac{NRMSE_i - NRMSE_{\text{AHF}}}{NRMSE_{\text{AHF}}} \times 100\% \tag{31}$$

The relative difference in NMAE (e^{NMAE}) of each surrogate is given by

$$e_i^{\text{NMAE}} = \frac{NMAE_i - NMAE_{\text{AHF}}}{NMAE_{\text{AHF}}} \times 100\% \tag{32}$$

where $i = 1, 2, 3, 4$ which represents QRSM, RBF, E-RBF, and Kriging, separately; $NRMSE_i$ and $NMAE_i$ are the NRMSE and the NMAE values of the ith surrogate, respectively; and $NRMSE_{\text{AHF}}$ and $NMAE_{\text{AHF}}$ refer the NRMSE and the NMAE values of the AHF surrogate, respectively. Table 6 shows the relative differences.

From Table 4, we observe that the overall performance of the AHF surrogate is better than that of the component surrogates. The AHF method yields the least RMSE for all test functions. From Table 5, we observe that the AHF method has a lower MAE value than does other methods in most cases except Goldstein & Price and Hartmann-6 functions.

The percentage-contributions of component surrogate models in the AHF are shown in Fig. 7. The percentage-contribution (η_i) of the ith component surrogate is given by

$$\eta_i = \frac{\sum_{j=1}^{n_t} w_j^i}{\sum_{i=1}^{n_s} \sum_{j=1}^{n_t} w_j^i} \tag{33}$$

where n_s is the number of surrogate models combined into the AHF, n_t is the number of test points; w_j^i represents the weight of the ith surrogate model for the jth test point.

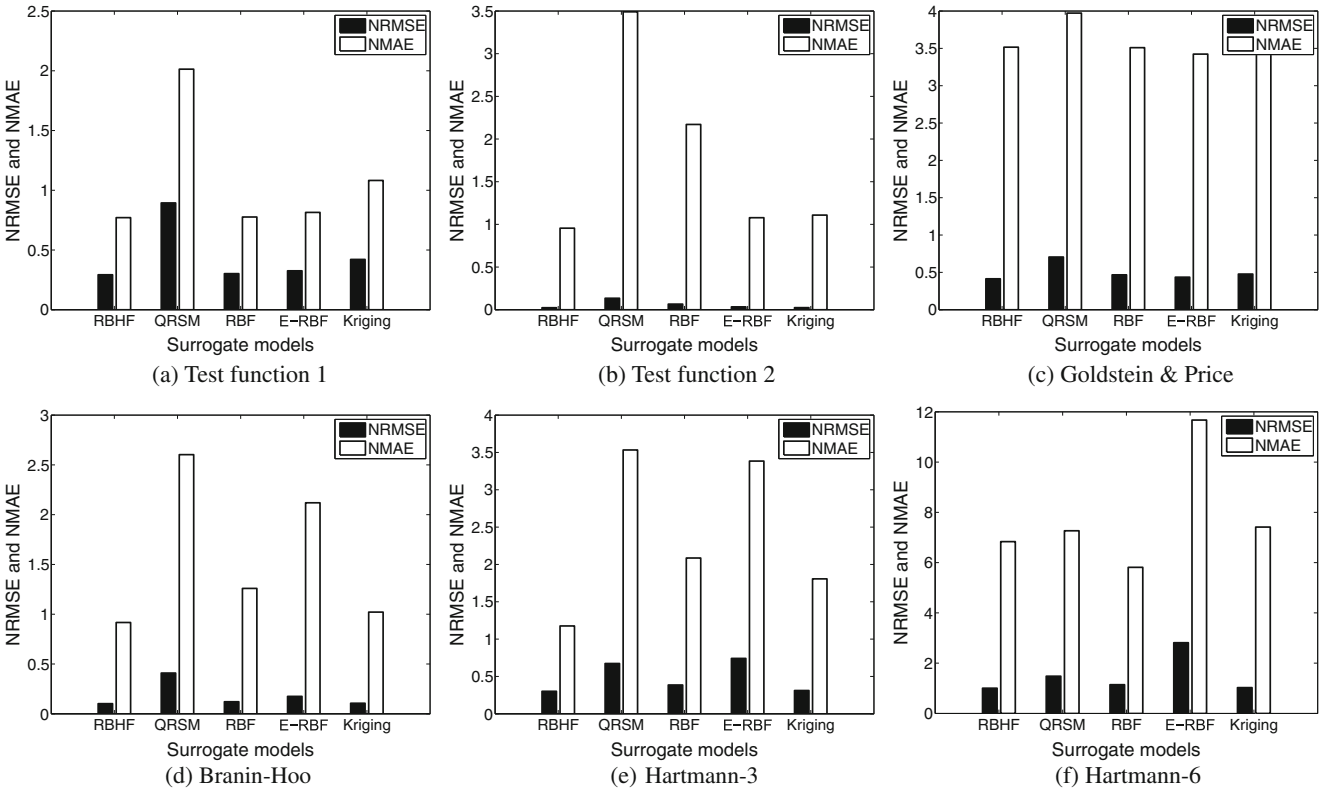
It is expected and desirable that a higher value of the RMSE (overall measure) for a component surrogate leads to a corresponding smaller contribution to the hybrid surrogate model. From this perspective, the RMSE shown in Fig. 6c and f, for the Goldstein & Price function and the Hartmann-6 function, respectively, are coherent with the corresponding percentage-contribution (illustrated in Fig. 7b and c,

Table 4 Comparison of the performances of each surrogate model using RMSE

Function	AHF		QRSM		RBF		E-RBF		Kriging	
	RMSE	NRMSE	RMSE	NRMSE	RMSE	NRMSE	RMSE	NRMSE	RMSE	NRMSE
Test function 1	**1.35e0**	**2.93e-1**	4.12e0	8.94e-1	1.39e0	3.02e-1	1.50e0	3.26e-1	1.94e0	4.21e-1
Test function 2	**2.99e0**	**2.36e-2**	1.71e1	1.35e-1	8.32e0	6.56e-2	4.24e0	3.34e-2	3.12e0	2.46e-2
Goldstein & Price	**1.49e5**	**4.14e-1**	2.54e5	7.06e-1	1.68e5	4.67e-1	1.57e5	4.36e-1	1.72e5	4.78e-1
Branin–Hoo	**7.99e0**	**1.02e-1**	3.21e1	4.09e-1	9.48e0	1.21e-1	1.37e1	1.75e-1	8.32e0	1.06e-1
Hartmann-3	**3.47e-1**	**3.02e-1**	7.75e-1	6.74e-1	4.44e-1	3.86e-1	8.53e-1	7.41e-1	3.59e-1	3.12e-1
Hartmann-6	**2.96e-1**	**1.00e0**	4.37e-1	1.48e0	3.37e-1	1.14e0	8.29e-1	2.81e0	3.03e-1	1.03e0
Wind farm problem	**1.10e-3**	**1.90e-3**	7.30e-3	1.35e-2	1.60e-1	2.94e-1	2.70e-3	5.00e-3	2.50e-3	4.60e-3

Table 5 Comparison of the performances of each surrogate model using MAE

Function	AHF		QRSM		RBF		E-RBF		Kriging	
	MAE	NMAE	MAE	NMAE	MAE	NMAE	MAE	NMAE	MAE	NMAE
Test function 1	**3.53e0**	**7.71e-1**	9.23e0	2.01e0	3.56e0	7.76e-1	3.73e0	8.15e-1	4.96e0	1.08e0
Test function 2	**1.68e1**	**9.56e-1**	6.12e1	3.49e0	3.80e1	2.17e0	1.89e1	1.08e0	1.94e1	1.11e0
Goldstein & Price	1.17e6	3.52e0	1.32e6	3.97e0	1.17e6	3.51e0	**1.14e6**	**3.42e0**	1.21e6	3.63e0
Branin–Hoo	**4.99e1**	**9.17e-1**	1.42e2	2.60e0	6.85e1	1.26e0	1.15e2	2.12e0	5.56e1	1.02e0
Hartmann-3	**1.02e0**	**1.18e0**	3.07e0	3.53e0	1.81e0	2.09e0	2.94e0	3.38e0	1.57e0	1.81e0
Hartmann-6	1.89e0	6.84e0	2.01e0	7.27e0	**1.60e0**	**5.81e0**	3.22e0	1.17e1	2.05e0	7.41e0
Wind farm problem	**2.60e-3**	**8.54e-2**	2.69e-2	8.98e-1	4.72e-1	1.58e1	8.20e-3	2.74e-1	8.40e-3	2.80e-1



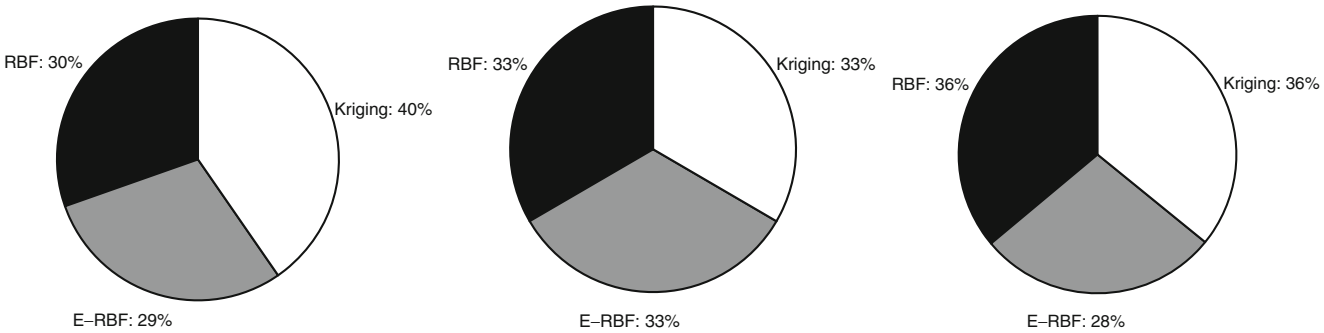
(a) Test function 1 (b) Test function 2 (c) Goldstein & Price

(d) Branin-Hoo (e) Hartmann-3 (f) Hartmann-6

Fig. 6 Comparison of the performances of each surrogate model

Table 6 Relative differences of NRMSE and NMAE by comparing AHF to other surrogates

Function	QRSM		RBF		E-RBF		Kriging	
	NRMSE (%)	NMAE (%)	NRMSE (%)	NMAE (%)	NRMSE (%)	NMAE (%)	NRMSE (%)	NMAE (%)
Test function 1	205.27	161.31	3.13	0.70	11.18	5.72	43.90	40.41
Test function 2	471.55	265.11	178.28	127.12	41.69	12.75	4.49	15.95
Goldstein & Price	70.36	12.96	12.69	−0.20	5.27	−2.65	15.34	3.24
Branin–Hoo	302.38	183.77	18.72	37.34	71.84	131.09	4.15	11.38
Hartmann-3	123.24	200.31	27.73	77.33	145.69	187.69	3.48	53.62
Hartmann-6	47.61	6.30	13.83	−14.98	180.18	70.75	2.37	8.47
Wind farm problem	596.26	951.68	15075.02	18359.46	156.86	220.21	135.41	228.20



(a) Test function 1 (b) Goldstein & Price (c) Hartmann-6

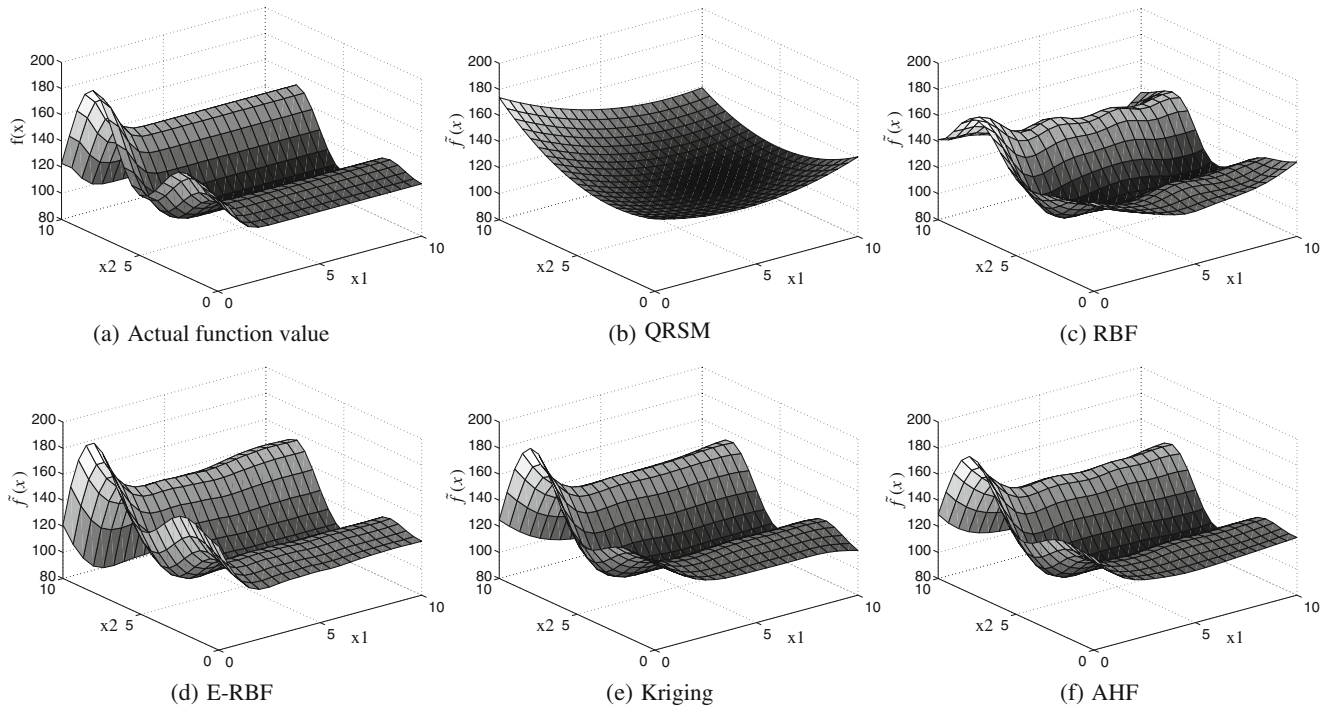
Fig. 7 Percentage-contribution of each surrogate model in the AHF

respectively). However, in the case of test function 1, Kriging has the highest percentage-contribution, in spite of yielding the largest RMSE. This observation is contrary to what is expected. This scenario might be attributed to the deterministic estimation of the trust region boundaries (quadratic approximations). Future research should pursue a stochastic estimation of the trust region boundaries (based on training point output).

The plots of the actual function and of the associated surrogate models for test function 1 are shown in Fig. 4. For the test functions with two input variables (test function 2, Goldstein & Price function, and Branin–Hoo function), the three-dimensional surface plots are shown in Figs. 8, 9, and 10, respectively. We observe that (i) all the three functions are highly nonlinear, and (ii) the AHF method can accurately represent the actual functions.

6.2 Wind farm power generation problem

The RMSE and the NRMSE estimated by each surrogate model for the wind farm power generation problem are shown in Table 4 and Fig. 11. It can be seen that the AHF performs the best, yielding the least RMSE and NRMSE



(a) Actual function value (b) QRSM (c) RBF

(d) E-RBF (e) Kriging (f) AHF

Fig. 8 Test function 2

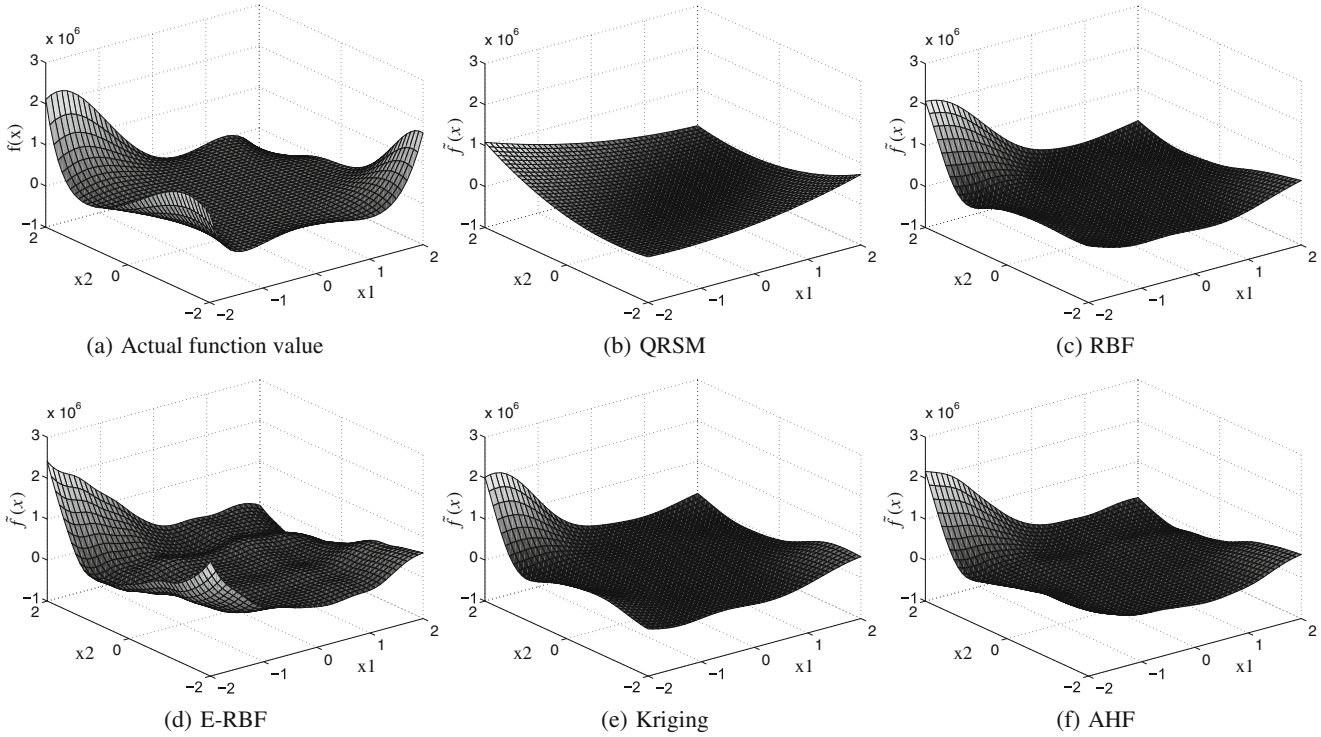


Fig. 9 Test function 3: Goldstein & Price function

values (shown in boldface). Likewise, AHF also has the least MAE and the NMAE values, which are shown in Table 5 and Fig. 11. Table 6 shows the relative differences of the estimated output values, which provides a ready comparison of the performance of AHF with that of its component surrogates. For this high dimensional problem, it can be seen that the relative differences are over 130%. Therefore, the hybrid surrogate proves to be a better choice

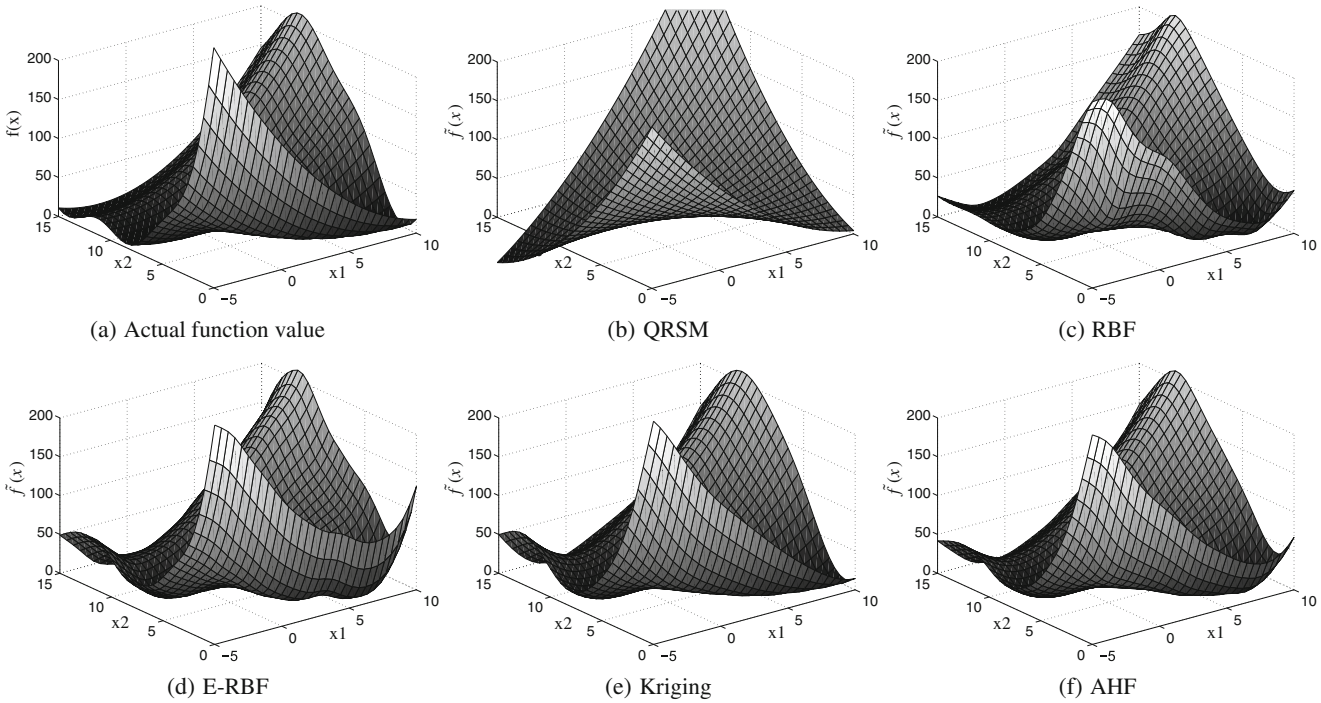


Fig. 10 Test function 4: Branin–Hoo function

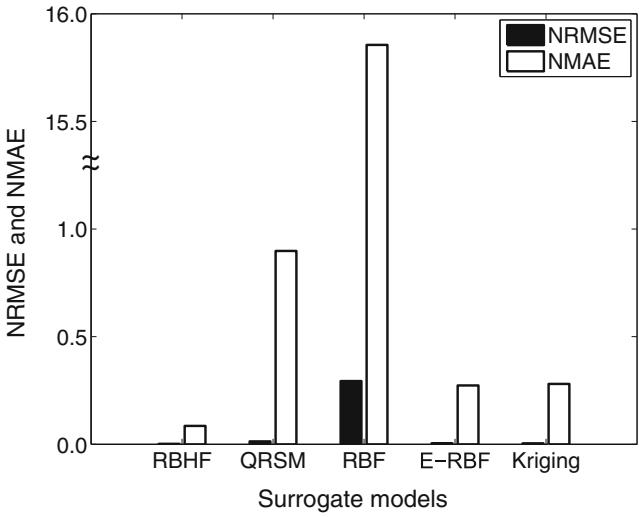


Fig. 11 NMSE and MAE for the wind farm problem

for representing the functional relationship in the wind farm optimization problem (Chowdhury et al. 2010, 2012), compared to the component surrogates.

7 Conclusion

This paper developed a generalized approach for constructing an effective surrogate model that can capture the global trend of the functional variation, while maintaining reasonable local accuracy. The Adaptive Hybrid Functions (AHF) method adaptively combines multiple surrogate models. In this method, we formulate a Crowding Distance-based Trust Region (CD-TR), and characterize the weight of each component surrogate model over the entire variable space. Four different surrogate models (QRSM, RBF, E-RBF and Kriging) are combined in the AHF method, in this paper. The performance of the surrogate is measured using two standard performance metrics: (i) Root Mean Squared Error (RMSE) and (ii) Maximum Absolute Error (MAE).

The effectiveness and the performance of the AHF surrogate model are tested on a series of standard examples, as well a real life problem-wind farm power generation model. Preliminary results show that the AHF can provide high fidelity approximations to complex and expensive functional relationships.

In the future, other error metrics that might better represent the performance over the entire design domain should be investigated. The application of the AHF to a more diverse set of test problems would more comprehensively assess the true potential of this novel method.

Acknowledgement Support from the National Science Foundation Awards CMMI-1100948, and CMMI-0946765 is gratefully acknowledged.

References

Acar E (2010) Various approaches for constructing an ensemble of metamodels using local measures. Struct Multidisc Optim 42(6):879–896

Acar E, Rais-Rohani M (2009) Ensemble of metamodels with optimized weight factors. Struct Multidisc Optim 37(3):279–294

Basudhar A, Missoum S (2008) Adaptive explicit decision functions for probabilistic design and optimization using support vector machines. Comput Struct 86(19–20):1904–1917

Bichon BJ, Eldred MS, Swiler LP, Mahadevan S, McFarland JM (2008) Efficient global reliability analysis for nonlinear implicit performance functions. AIAA J 46(10):2459–2468

Bichon BJ, Mahadevan S, Eldred MS (2009) Reliability-based design optimization using efficient global reliability analysis. In: 50th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics, and materials conference, Palm Springs, California

Bratley P, Fox B (1988) Algorithm 659: implementing Sobol's quasirandom sequence generator. ACM Trans Math Softw 14(1):88–100

Cherrie JB, Beatson RK, Newsam GN (2002) Fast evaluation of radial basis functions: methods for generalized multiquadrics in r^n. SIAM J Sci Comput 23(5):1549–1571

Chowdhury S, Messac A, Zhang J, Castillo L, Lebron J (2010) Optimizing the unrestricted placement of turbines of differing rotor diameters in a wind farm for maximum power generation. In: ASME 2010 international design engineering technical conferences (IDETC), Montreal, Canada

Chowdhury S, Zhang J, Messac A, Castillo L (2012) Unrestricted wind farm layout optimization (uwflo): investigating key factors influencing the maximum power generation. Renew Energy 38(1):16–30

Clarke S, Griebsch J, Simpson T (2005) Analysis of support vector regression for approximation of complex engineering analyses. J Mech Des 127(6):1077–1087

Cressie N (1993) Statistics for spatial data. Wiley, New York

Deb K (2001) Multi-objective optimization using evolutionary algorithms. Wiley, New York

Duda R, Hart P, Stork D (2000) Pattern classification, 2nd edn. Wiley-Interscience, New York

Forrester A, Keane A (2009) Recent advances in surrogate-based optimization. Prog Aerosp Sci 45(1–3):50–79

Forrester A, Sobester A, Keane A (2008) Engineering design via surrogate modelling: a practical guide. Wiley, New York

Giunta A, Watson L (1998) A comparison of approximation modeling techniques: polynomial versus interpolating models. Tech Rep AIAA-98-4758

Goel T, Haftka R, Shyy W, Queipo N (2007) Ensemble of surrogates. Struct Multidisc Optim 33(3):199–216

Hardy RL (1971) Multiquadric equations of topography and other irregular surfaces. J Geophys Res 76:1905–1915

Huang D, Allen TT, Notz WI, Zeng N (2006) Global optimization of stochastic black-box systems via sequential kriging meta-models. J Glob Optim 34(3):441–466

Hussain MF, Barton RR, Joshi SB (2002) Metamodeling: radial basis functions, versus polynomials. Eur J Oper Res 138(1):142–154

Jin R, Chen W, Simpson T (2001) Comparative studies of metamodelling techniques under multiple modelling criteria. Struct Multidisc Optim 23(1):1–13

Jones D, Schonlau M, Welch W (1998) Efficient global optimization of expensive black-box functions. J Glob Optim 13(4):455–492

Lophaven S, Nielsen H, Sondergaard J (2002) Dace—a matlab Kriging toolbox, version 2.0. Tech Rep Informatics and mathematical modelling report IMM-REP-2002-12, Technical University of Denmark

McKay M, Conover W, Beckman R (1979) A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 21(2):239–245

Mullur A, Messac A (2005) Extended radial basis functions: more flexible and effective metamodeling. AIAA J 43(6):1306–1315

Mullur A, Messac A (2006) Metamodeling using extended radial basis functions: a comparative approach. Eng Comput 21(3):203–217

Myers R, Montgomery D (2002) Response surface methodology: process and product optimization using designed experiments, 2nd edn. Wiley-Interscience, New York

NDAWN (2010) The North Dakota Agricultural Weather Network. http://ndawn.ndsu.nodak.edu/

Queipo N, Haftka R, Shyy W, Goel T, Vaidyanathan R, Tucker P (2005) Surrogate-based analysis and optimization. Prog Aerosp Sci 41(1):1–28

Sakata S, Ashida F, Zako M (2003) Structural optimization using Kriging approximation. Comput Methods Appl Mech Eng 192(7–8):923–939

Sanchez E, Pintos S, Queipo N (2008) Toward an optimal ensemble of kernel-based approximations with engineering applications. Struct Multidisc Optim 36(3):247–261

Simpson T (1998) A concept exploration method for product family design. PhD thesis, Georgia Institute of Technology

Simpson T, Toropov V, Balabanov V, Viana F (2008) Design and analysis of computer experiments in multidisciplinary design optimization: a review of how far we have come—or not. In: 12th AIAA/ISSMO multidisciplinary analysis and optimization conference, Victoria, Canada

Sobol I (1976) Uniformly distributed sequences with an additional uniform property. USSR Comput Math Math Phys 16(5):236–242

Vapnik V (1995) The nature of statistical learning theory. Springer, New York

Viana F, Haftka R, Steffen V (2009) Multiple surrogates: how cross-validation errors can help us to obtain the best predictor. Struct Multidisc Optim 39(4):439–457

Viana FAC, Haftka RT, Watson LT (2010) Why not run the efficient global optimization algorithm with multiple surrogates? In: 51st AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics, and materials conference, Orlando, Florida

Wang G, Shan S (2007) Review of metamodeling techniques in support of engineering design optimization. J Mech Des 129(4):370–380

Yegnanarayana B (2004) Artificial neural networks. PHI Learning Pvt Ltd

Zerpa L, Queipo N, Pintos S, Salager J (2005) An optimization methodology of alkalineŰsurfactantŰpolymer flooding processes using field scale numerical simulation and multiple surrogates. J Pet Sci Eng 47(3–4):197–208

Zhang J, Chowdhury S, Messac A, Castillo L (2010a) Economic evaluation of wind farms based on cost of energy optimization. In: 13th AIAA/ISSMO multidisciplinary analysis optimization conference, Fort Worth, Texas

Zhang J, Chowdhury S, Messac A, Castillo L, Lebron J (2010b) Response surface based cost model for onshore wind farms using extended radial basis functions. In: ASME 2010 international design engineering technical conferences (IDETC), Montreal, Canada

Zhang JQ, Messac A, Zhang J, Chowdhury S (2010c) Comparison of surrogate models used for adaptive optimal control of active thermoelectric windows. In: 13th AIAA/ISSMO multidisciplinary analysis optimization conference, Fort Worth, Texas

Zhou X, Ma Y, Li X (2011) Ensemble of surrogates with recursive arithmetic average. Struct Multidisc Optim. doi:10.1007/s00158-011-0655-6