AIAA 2011-2152

52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, 4 - 7 April 2011, Denver, Colorado

SDM 2011 Student Papers Competition
A New Robust Surrogate Model: Reliability Based Hybrid Functions

Jie Zhang and Souma Chowdhury*
Rensselaer Polytechnic Institute, Troy, New York 12180

Achille Messac†
Syracuse University, Syracuse, NY, 13244

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 aggregated, offers a robust solution. In this paper, we develop a new high fidelity surrogate modeling technique that we call the Reliability Based Hybrid Functions (RBHF). The RBHF 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 reliability measure 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 and the local deviations. In this paper, the RBHF integrates 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 RBHF is applied to standard test problems. 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

I. Introduction

The need to quantify economic and engineering performance of complex products 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 a wide acceptance from the design community. Surrogate modeling is concerned with the construction of approximation models to estimate the 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 the optimization of

*Doctoral Student, Multidisciplinary Design and Optimization Laboratory, Department of Mechanical, Aerospace and Nuclear Engineering, AIAA student member

†Distinguished Professor and Department Chair. Department of Mechanical and Aerospace Engineering, AIAA Lifetime Fellow. Corresponding author. Email: messac@syr.edu

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 accurately represents the original function over a given design domain".

Several generalized function estimation techniques have been developed over the past few decades - e.g. Kriging, polynomial response surfaces, and radial basis function based response surfaces. Advanced versions of these function estimation methods have been frequently reported in the literature, particularly that pertaining to the Multidisciplinary Design Optimization (MDO) community; in this community, function estimation is more popularly known as *surrogate modeling*. However, in the MDO field, the development of surrogate modeling is driven by the need to avoid the high computational expenses of estimating 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 a robust 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 II reviews the existing surrogate modeling methods; Section III presents the motivation and objective of this research; the Reliability Based Hybrid Functions (RBHF) is formulated in Section IV; numerical experiments and results are elaborated in Section V and Section VI.

II. Surrogate Modeling Review

A wide variety of surrogate modeling techniques have been reported in the literature, such as: (*i*) Polynomial Response Surface Model (PRSM),[1] (*ii*) Kriging,[2–5] (*iii*) Radial Basis Functions (RBF),[6–9] (*iv*) Extended Radial Basis Functions (E-RBF),[10–14] (*v*) Artificial Neural Networks (ANN),[15,16] and (*vi*) Support Vector Regression (SVR).[15,17–19] In the literature, the accuracy and the robustness of various surrogate models for linear, nonlinear, smooth, and noisy responses have also been investigated.[20–23]

PRSM is a statistical tool, 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.[20,21] PRSM can generally capture the global trend and involve a desirably small set of parameters (unknown coefficients). However, PRSM is often not adequate for capturing the local accuracy around the training points. The challenge of exact fitting has inspired researchers to explore the so-called nonparametric surrogate modeling techniques, which provide an interpolating surface through the entire training data set. Nonparametric surrogate modeling techniques offer critical advantages over the traditional PRSM, such as the ease of extending the estimated function to higher dimensions and the greater accuracy in the local region close to the training points. Kriging, RBF and E-RBF are among the most popular nonparametric surrogate modeling techniques.

In surrogate-based modeling of complex problems, the common practice is (i) to construct one or more surrogate models and (ii) to select the one with the best performance. This approach does not exploit the full advantage of the resources devoted to constructing the surrogate model.

More recently, researchers have presented the development of a combination of different approximate models into a single hybrid model for developing weighted average surrogates.[24–28] Zepra et al.[24] 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 has better performance than individual surrogates. Goel et al.[25] 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[27] 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 model.

III. Research Objectives

Building on the previous research, this paper offers an innovative approach to develop a robust surrogate model, the Reliability Based Hybrid Functions (RBHF).

Each surrogate modeling methodology (or the underlying basis functions) has its own benefits and limitations; hence, they are expected to present widely different levels of numerical fidelity for different types of problems. We have hypothesized that, an approach that can appropriately combine the favorable characteristics of the different surrogate modeling methods (a *hybrid surrogate*) would be able to address a broad scope of problems (that demand function estimation). Assuming that the designer/researcher does not have a 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 (often contradicting) objectives of surrogate modeling can be divided into two parts:

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

Simultaneous fulfillment of the above objectives presents significant challenges to the surrogate modeling approaches. An effort that can improve the overall fidelity of the estimated function, without additional functional evaluations (compared to individual surrogate models) would be particularly helpful. We believe that a hybrid surrogate model has the potential to address the above-stated challenges.

IV. Reliability Based Hybrid Functions (RBHF)

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* means that the surrogate can capture the global trend for a given set of simulated or experimental data; errors between the estimated and the actual function might 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 RBHF methodology developed in this paper seeks to simultaneously capture the global and the local accuracy. The RBHF surrogate integrates (*i*) Quadratic Response Surface Method (QRSM), (*ii*) Radial Basis Functions (RBFs), (*iii*) Extended Radial Basis Functions (E-RBF), and (*iv*) Kriging.

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

1. Determination of a trust region: numerical bounds of the estimated parameter (output) as a function of the independent parameters (input vector) over the feasible input space.
2. Characterization of the local reliability (using probability distribution functions) of the estimated function value, and the representation of the corresponding distribution parameters as functions of the input vector.
3. Generation of different surrogate models (component surrogates), and weighted aggregate of the estimated function value based on the local reliability of the individual surrogates (modeled in the previous step).

The primary objective of this paper is to lay the foundation for such a reliability-based hybrid surrogate model. At the same time, we also develop preliminary strategies to implement this 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 RBHF surrogate model, is illustrated in Fig. 1. In the representation given by D, x_j^i is the j^{th} dimension of the input vector representing the i^{th} training point, and y^i is the corresponding output; n_d is the dimension of the input variable, and n_p represents the number of training data points.

We will use a two-dimensional scenario to illustrate the steps of the RBHF formulation, for the sake of presentation simplicity. The function used for this purpose is test function 1; details of this function is provided in Section V.

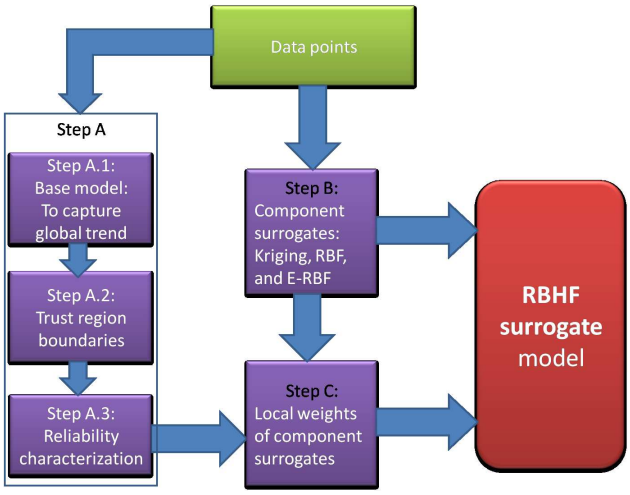


Figure 1. The framework of the RBHF surrogate model

A. 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 could capture the global trend of training points, thereby seeking to improve 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 to use other (typically monotonic) smooth functions as well. A typical QRSM can be represented as

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

where the x_i's are the input parameters and the a_{ij}'s are the unknown coefficients determined by the least squares approach.

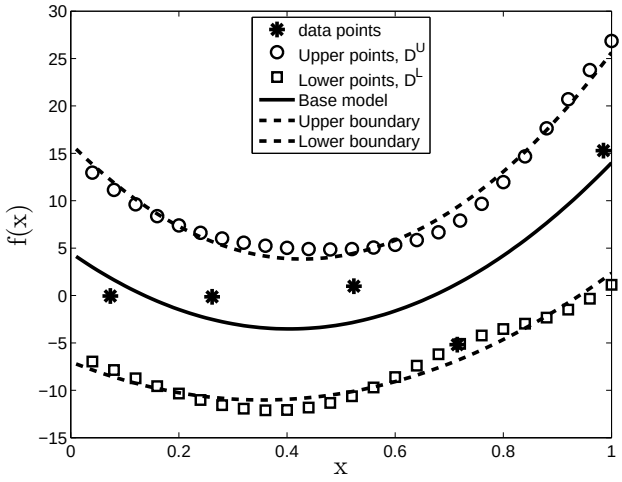


Figure 2. Base and boundary models of RBHF surrogate

B. 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 are adaptively constructed based on the base model. This approach seeks to significantly reduce the "deviation of the final surrogate from the actual function", in respective of what locally accurate surrogate models are aggregated in the third step. The final hybrid surrogate model adaptively aggregates three nonparametric surrogate modeling techniques (Kriging, RBF, E-RBF) in the CD-TR, which are known to capture the local accuracy.

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.[29] In this paper, crowding distance is used to evaluate the density of training points surrounding any point on the base model (solid line in Fig. 2). Ideally,

1. Large crowding distance value of a point reflects low sample density (fewer points around that point), and the reliability of the surrogate is expected to be relatively lower around that point. Thus, we need wider boundaries at that point.

2. Small crowding distance value of a point reflects high sample density (more points around that point), and the reliability of the surrogate is expected to be relatively higher around that point. Then, tighter boundaries can be employed.

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

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

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

$$\rho^i = \frac{1}{CD^i} \tag{2}$$

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

$$\alpha_i = \frac{\max(\rho) - \rho^i}{\max(\rho) - \min(\rho)} \tag{3}$$

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

$$d^i = (1 + \alpha_i) \times \max_{j \in D} \left| \tilde{f}_{qrsm}(x^j) - y^j \right| \tag{4}$$

where D represents the training data set. In Eq. 4, $\tilde{f}_{qrsm}(x^j)$ is the estimated output value of the j^{th} training point using QRSM; $\left| \tilde{f}_{qrsm}(x^j) - y^j \right|$ is the distance from the j^{th} training point to the base model along the direction of the output axis. Subsequently, we could obtain two sets of points, D^U and D^L, for constructing the two boundaries, as expressed by

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

Again, 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 of individual component surrogates in the following step.

It is important to note that the CD-TR estimation is particulary 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 (simulation-based), the initial sample data is expected to be relatively evenly distributed; significant variation in crowding distance might not be observed.

C. Step A.3: Estimation of Probability Distributions

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

The function estimation is performed between the two boundary surfaces, using a local reliability technique. The uncertainty in the function estimated at a location in the input variable domain is modeled using a *probability density function (pdf)*. This distribution is expressed as a function of the output parameter. The corresponding typical pdf coefficients are represented as functions of the input vector, thereby characterizing the reliability of the estimated function over the entire input domain.

An example of the local distribution for a 2D problem (test function 1) is shown in Fig. 3. There is a scope to implement a wide set of distribution functions to accomplish the probability approximation. In this paper, a fixed amplitude Gaussian pdf is adopted to represent the reliability of the estimated function value. Future research should investigate the use of other distribution functions. Importantly, the RBHF method has the flexibility to select any other probability distribution functions. The Gaussian pdf used here is expressed as

$$P(z) = a \exp\left[-\frac{(z-\mu)^2}{2\sigma^2}\right] \tag{5}$$

where the coefficients a, μ and σ represent the amplitude, the mean and the standard deviation of the distribution, respectively.

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 function reliability (pdf) is 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 amplitude is set to be equal to one ($a = 1$) in this paper. The output value of a training point does not necessarily occur midway between the two boundaries. In order to ensure the continuity of the distribution function, we divide the function into two parts, with distinct standard deviations and the same mean. Then, we can represent the distribution as

$$P(x^i) = \begin{cases} a \exp\left\{-\frac{\left[y(x^i)-\mu(x^i)\right]^2}{2\sigma_1^2(x^i)}\right\} & \text{if } 0 \le y(x^i) \le \mu(x^i) \\ a \exp\left\{-\frac{\left[y(x^i)-\mu(x^i)\right]^2}{2\sigma_2^2(x^i)}\right\} & \text{if } \mu(x^i) \le y(x^i) \le 1 \end{cases} \tag{6}$$

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

$$\begin{aligned} \sigma_1(x^i) &= \frac{\Delta z_{10}(x^i)}{2\sqrt{2\ln 10}} = \frac{2\left[\mu(x^i) - f_L^i(x^i)\right]}{2\sqrt{2\ln 10}} \\ &= \frac{2\mu(x^i)}{2\sqrt{2\ln 10}} = \frac{\mu(x^i)}{\sqrt{2\ln 10}} \qquad , and \end{aligned} \tag{7}$$

$$\begin{aligned} \sigma_2(x^i) &= \frac{\Delta z_{10}(x^i)}{2\sqrt{2\ln 10}} = \frac{2\left[f_U^i(x^i) - \mu(x^i)\right]}{2\sqrt{2\ln 10}} \\ &= \frac{2\left[1 - \mu(x^i)\right]}{2\sqrt{2\ln 10}} = \frac{1-\mu(x^i)}{\sqrt{2\ln 10}} \qquad , where \end{aligned} \tag{8}$$

$$P(\mu \pm 0.5\Delta z_{10}) = \frac{1}{10} \tag{9}$$

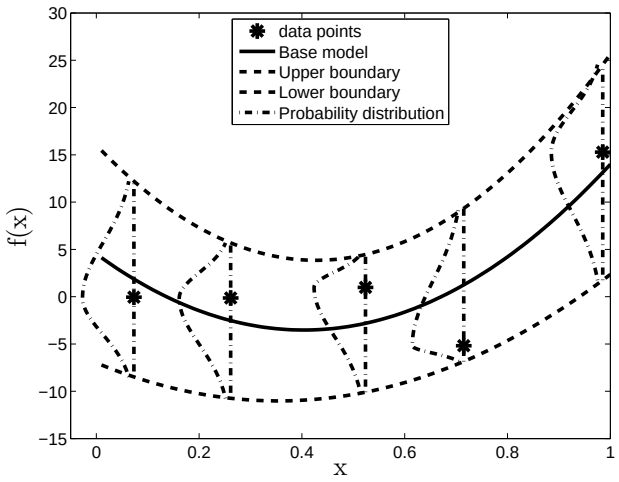


Figure 3. Probability distribution evaluation of the RBHF surrogate

From Eq. 6, we can determine the reliability coefficients $\mu(x^i)$ for the i^{th} 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 reliability can be estimated using the RMSM method. In the following, we develop each component surrogate and determine the weight of that component for the hybrid model.

D. Step B: Creation of Surrogate Models Using Existing Methods

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

1. Kriging

Kriging[2–4] is an approach to approximate irregular data. The kriging approximation function consists of two parts: (i) a global trend function, and (ii) a functional 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.[4] The general form of the kriging surrogate model is given by:[5]

$$\tilde{f}(x) = P(x) + Z(x) \tag{10}$$

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

$$COV[Z(x^i), Z(x^j)] = \sigma_z^2 R_{ij} \tag{11}$$

where R_{ij} is the correlation function between the i^{th} and the j^{th} data points; and σ_z^2 is the process variance. In the present paper, a gaussian function is used as the correlation function, defined as

$$R(x^i, x^j) = R_{ij} = exp\left\{-\Sigma_{k=1}^{m}\theta_k\left(x_k^i - x_k^j\right)^2\right\} \tag{12}$$

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

2. Radial Basis Functions

The idea of using Radial Basis Functions (RBF) as approximation functions was introduced by Hardy[6] in 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.[7–9]

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,[6,8] which is defined as

$$\psi(r) = \sqrt{r^2 + c^2} \tag{13}$$

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

$$\tilde{f}(x) = \sum_{i=1}^{n_p} \sigma_i \psi\left(\|x - x^i\|\right) \tag{14}$$

where $\sigma_i^{'}s$ 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 (14) can be solved using the pseudo inverse method.

3. Extended Radial Basis Functions

The Extended Radial Basis Functions (E-RBF)[10] 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 j^{th} dimension. The N-RBF for the i^{th} data point and the j^{th} dimension is denoted by ϕ_{ij}. It is composed of three distinct components, as given by

$$\phi_{ij}(\xi_j^i) = \alpha_{ij}^L \phi^L(\xi_j^i) + \alpha_{ij}^R \phi^R(\xi_j^i) + \beta_{ij}\phi^\beta(\xi_j^i) \tag{15}$$

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

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

$$\tilde{f}(x) = \sum_{i=1}^{n_p} \sigma_i \psi\left(\|x - x^i\|\right) + \sum_{i=1}^{n_p}\sum_{j=1}^{m} \left\{\alpha_{ij}^L \phi^L(\xi_j^i) + \alpha_{ij}^R \phi^R(\xi_j^i) + \beta_{ij}\phi^\beta(\xi_j^i)\right\} \tag{16}$$

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, the total number of coefficients to be determined is given by $(3m+1)n_p$. Two methods that can be used to solve Eqn. (16) are (i) linear programming and (ii) pseudo inverse. Mullur and Messac[10] provides the details of the E-RBF approach.

E. Step C: Determining Local Weights of Component Surrogates

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

$$\tilde{f}_{rbhf} = \sum_{i=1}^{n_s} w_i(x)\tilde{f}_i(x) \tag{17}$$

where n_s is the number of component surrogates integrated into the RBHF, 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 reliability, which is given by

$$w_i(x) = \frac{P_i(x)}{\sum_{i=1}^{n_s} P_i(x)} \tag{18}$$

where $P_i(x)$ is the probability value of the i^{th} surrogate for point x.

In this paper, three surrogates (RBF, E-RBF and Kriging) are integrated into the RBHF ($n_s = 3$), and the final RBHF 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.

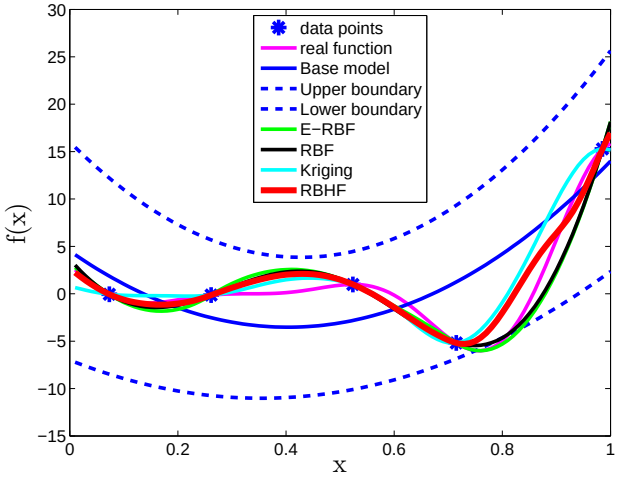


Figure 4. Reliability Based Hybrid Function

V. Numerical Examples

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

A. Benchmark Problems

The performance of the RBHF is illustrated using the following analytical benchmark problems: (*i*) 1-variable function,[30] (*ii*) 2-variable function,[10] (*iii*) Goldstein & Price function,[25] (*iv*) Branin-Hoo function,[25] (*v*) Hartmann function with 3 variables,[25] and (*vi*) Hartmann function with 6 variables.[25] The expressions of these problems are summarized as follows:

Test Function 1: 1-Variable Function

$$f(x) = (6x_1 - 2)^2 \sin\left[2\,(6x_1 - 2)\right] \tag{19}$$
$$where \quad x_1 \in [0 \;\; 1]$$

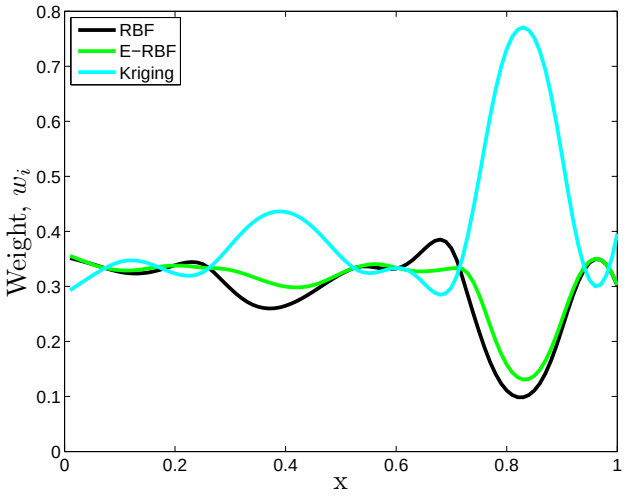


Figure 5. The weight values of each component surrogate

Test Function 2: 2-Variable Function

$$f(x) = [30 + x_1 sin(x_1)] \times \left[4 + exp\left(-x_2^2\right)\right] \tag{20}$$
$$where \quad x_1 \in [0 \;\; 10], \quad x_2 \in [0 \;\; 10]$$

Test Function 3: Goldstein & Price Function

$$f(x) = [1 + (x_1 + x_2 + 1)^2 \times \left(19 - 14x_1 + 13x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2\right)] \times \tag{21}$$
$$[30 + (2x_1 - 3x_2)^2 \times \left(18 - 32x_1 + 12x_1^2 - 48x_2 - 36x_1x_2 + 27x_2^2\right)]$$
$$where \quad x_1 \in [-2 \;\; 2], \quad x_2 \in [-2 \;\; 2]$$

Test Function 4: Branin-Hoo Function

$$f(x) = \left(x_2 - \frac{5.1x_1^2}{4\pi^2} + \frac{5x_1}{\pi} - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right) cos(x_1) + 10 \tag{22}$$
$$where \quad x_1 \in [-5 \;\; 10], \quad x_2 \in [0 \;\; 15]$$

Test Function 5 and 6: Hartmann Function

$$f(x) = -\sum_{i=1}^{4} c_i \; exp\left\{-\sum_{j=1}^{n} A_{ij}\left(x_j - P_{ij}\right)^2\right\} \tag{23}$$
$$where \quad x = (x_1 \;\; x_2 \;\; \ldots \;\; x_n) \quad x_i \in [0 \;\; 1]$$

Two instances of this problem are considered based on the number of design variables, (i) Hartmann-3 with 3 input variables, and (i) Hartmann-6 with 6 input variables. 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

$$A = \begin{bmatrix} 3.0 & 10 & 30 \\ 0.1 & 10 & 35 \\ 3.0 & 10 & 30 \\ 0.1 & 10 & 35 \end{bmatrix}, \quad P = \begin{bmatrix} 0.3689 & 0.1170 & 0.2673 \\ 0.4699 & 0.4387 & 0.7470 \\ 0.1091 & 0.8732 & 0.5547 \\ 0.03815 & 0.5743 & 0.8828 \end{bmatrix}$$

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

$$A = \begin{bmatrix} 10.0 & 3.0 & 17.0 & 3.5 & 1.7 & 8.0 \\ 0.05 & 10.0 & 17.0 & 0.1 & 8.0 & 14.0 \\ 3.0 & 3.5 & 1.7 & 10.0 & 17.0 & 8.0 \\ 17.0 & 8.0 & 0.05 & 10.0 & 0.1 & 14.0 \end{bmatrix}, \quad P = \begin{bmatrix} 0.1312 & 0.1696 & 0.5569 & 0.0124 & 0.8283 & 0.5886 \\ 0.2329 & 0.4135 & 0.8307 & 0.3736 & 0.1004 & 0.9991 \\ 0.2348 & 0.1451 & 0.3522 & 0.2883 & 0.3047 & 0.6650 \\ 0.4047 & 0.8828 & 0.8732 & 0.5743 & 0.1091 & 0.0381 \end{bmatrix}$$

B. 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. Latin hypercube sampling method is adopted in this paper. Latin hypercube sampling, used for the benchmark problems, is a strategy for generating random sample points, which ensures a practically uniform representation of the entire variable domain.[31] 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 n_p values in each column are randomly selected - one from each of the intervals, $(0, 1/n_p),\ (1/n_p, 2/n_p), \ldots, (1 - 1/n_p, 1)$.[31]

C. 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.

Table 2. Parameter selection for the E-RBF method

Parameter	λ	c	t
Value	4.75	0.9	2

For the Kriging method used in this paper, we have used an efficient MATLAB implementation, DACE (design and analysis of computer experiments), developed by Lophaven et al.[32] The bounds on the correlation parameters in the nonlinear optimization, θ_l and θ_u, are selected as 0.1 and 20. Under the kriging approach, the order of the global polynomial trend function was specified to be zero.

D. Performance Criteria

The overall performance of the surrogates is evaluated using two standard performance metrics: (i) Root Mean Squared Error (RMSE),[7,20] which provides a global error measure over the entire design domain, and (ii) Maximum Absolute Error (MAE),[11,21] 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.

1. Root Mean Squared Error (RMSE)

The RMSE is given by

$$RMSE = \sqrt{\frac{1}{n_t}\sum_{k=1}^{n_t}\left(f(x^k) - \tilde{f}(x^k)\right)^2} \tag{24}$$

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. n_t is the number of test points chosen for evaluating the error measure. The Normalized Root Mean Squared Error (NRMSE) is given by

$$NRMSE = \sqrt{\frac{\sum_{k=1}^{n_t}\left(f(x^k) - \tilde{f}(x^k)\right)^2}{\sum_{k=1}^{n_t}\left(f(x^k)\right)^2}} \tag{25}$$

2. Maximum Absolute Error (MAE)

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

$$MAE = \max_k \left|f(x^k) - \tilde{f}(x^k)\right| \tag{26}$$

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

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

E. 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.

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

VI. Results and Discussion

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 shown in boldface in the table as well. The comparison of the performance of the individual surrogates and the RBHF is illustrated through bar diagrams in Fig. 6.

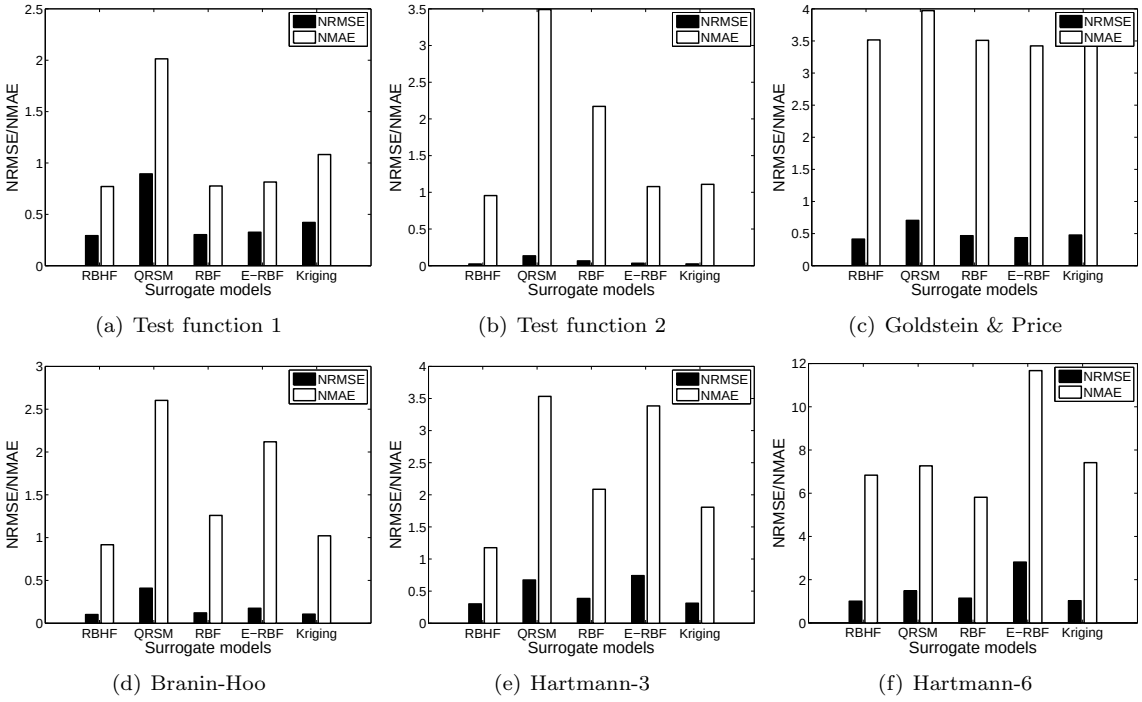
Table 4. Comparison of the performances of each surrogate model using RMSE

	RBHF		QRSM		RBF		E-RBF		Kriging	
Function	RMSE	NRMSE	RMSE	NRMSE	RMSE	NRMSE	RMSE	NRMSE	RMSE	NRMSE
Test function 1	**1.3503**	**0.2927**	4.1221	0.8936	1.3926	0.3019	1.5013	0.3255	1.9430	0.4212
Test function 2	**2.9904**	**0.0236**	17.0914	0.1347	8.3215	0.0656	4.2370	0.0334	3.1248	0.0246
Goldstein & Price	**148807**	**0.4144**	253513	0.7060	167686	0.4670	156644	0.4362	171640	0.4780
Branin-Hoo	**7.9852**	**0.1017**	32.1308	0.4092	9.4800	0.1207	13.7220	0.1748	8.3166	0.1059
Hartmann-3	**0.3472**	**0.3018**	0.7751	0.6737	0.4435	0.3855	0.8530	0.7414	0.3593	0.3123
Hartmann-6	**0.2957**	**1.0037**	0.4365	1.4815	0.3366	1.1425	0.8285	2.8121	0.3027	1.0275

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

	RBHF		QRSM		RBF		E-RBF		Kriging	
Function	MAE	NMAE	MAE	NMAE	MAE	NMAE	MAE	NMAE	MAE	NMAE
Test function 1	**3.5316**	**0.7705**	9.2287	2.0134	3.5562	0.7759	3.7337	0.8146	4.9589	1.0819
Test function 2	**16.7527**	**0.9560**	61.1657	3.4903	38.0485	2.1712	18.8892	1.0779	19.4252	1.1085
Goldstein & Price	1168450	3.5161	1319881	3.9718	**1166080**	**3.5090**	1137492	3.4230	1206256	3.6299
Branin-Hoo	**49.9408**	**0.9171**	141.7171	2.6026	68.5895	1.2596	115.4078	2.1194	55.6217	1.0215
Hartmann-3	**1.0226**	**1.1764**	3.0711	3.5329	1.8135	2.0861	2.9419	3.3843	1.5710	1.8072
Hartmann-6	1.8869	6.8354	2.0057	7.2657	**1.6043**	**5.8117**	3.2219	11.6714	2.0468	7.4146

From Table 4, we observe that the overall performance of the RBHF surrogate is better than the component surrogates. The RBHF method yields the least RMSE for all the test functions. From Table 5,



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

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

Figure 6. Comparison of the performances of each surrogate model

we observe that the RBHF method has a lower MAE value than other methods in most cases except test functions 3 and 6 (Goldstein & Price and Hartmann-6).

The *percentage-contribution* of component surrogate models in the RBHF are shown in Fig. 7. The *percentage-contribution* (η_i) of the i^{th} component surrogate is given by

$$\eta_i = \frac{\sum_{j=1}^{n_t} w_j^i}{\sum_{i=1}^{n_s} \sum_{j=1}^{n_t} w_j^i} \tag{28}$$

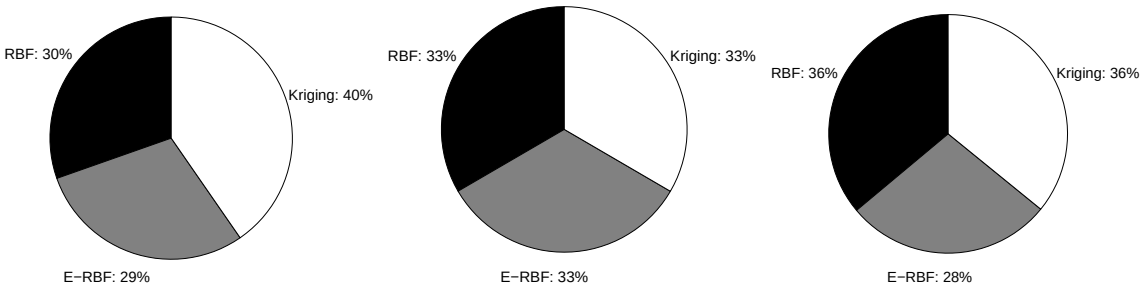
where n_s is the number of surrogate models integrated into the RBHF, n_t is the number of test points; w_j^i represents the weight of the i^{th} surrogate model for the j^{th} test point.

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 Figs. 6(c) and 6(f), for the Goldstein & Price function and the Hartmann-6 function, respectively, are coherent with the corresponding *percentage-contribution* (illustrated in Figs. 7(b) and 7(c), 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 discrepancy 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) to address this issue.

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, 3 (Goldstein & Price), and 4 (Branin-Hoo)), 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 functions, and (ii) the RBHF method can accurately represent the actual functions.

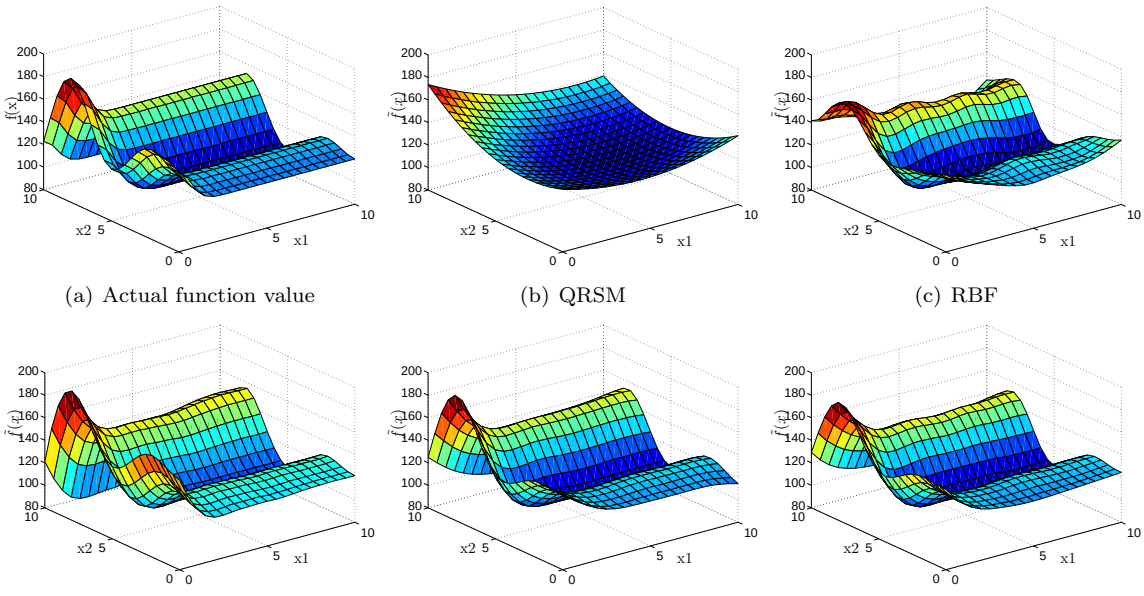
VII. Conclusion

This paper develops a generalized approach for constructing a robust surrogate model that can capture the global trend of the functional variation and accomplish reasonable local accuracy as well. The RBHF method adaptively combines multiple surrogate models. In this method, we formulate a Crowding Distance-



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

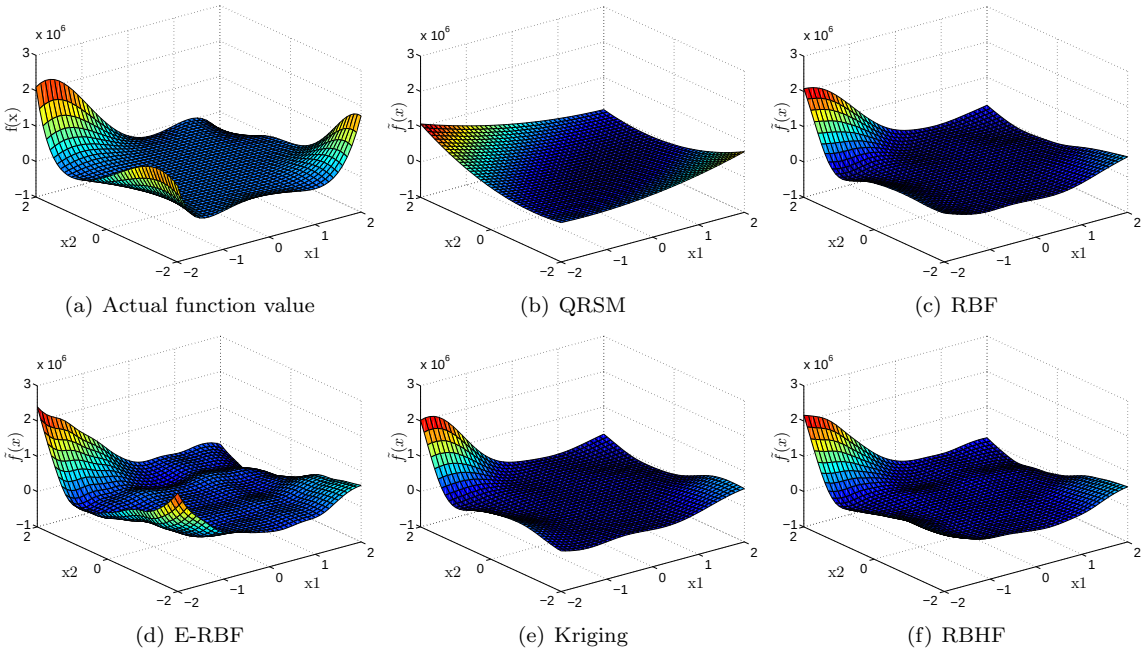
Figure 7. Percentage of each surrogate model in the RBHF



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

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

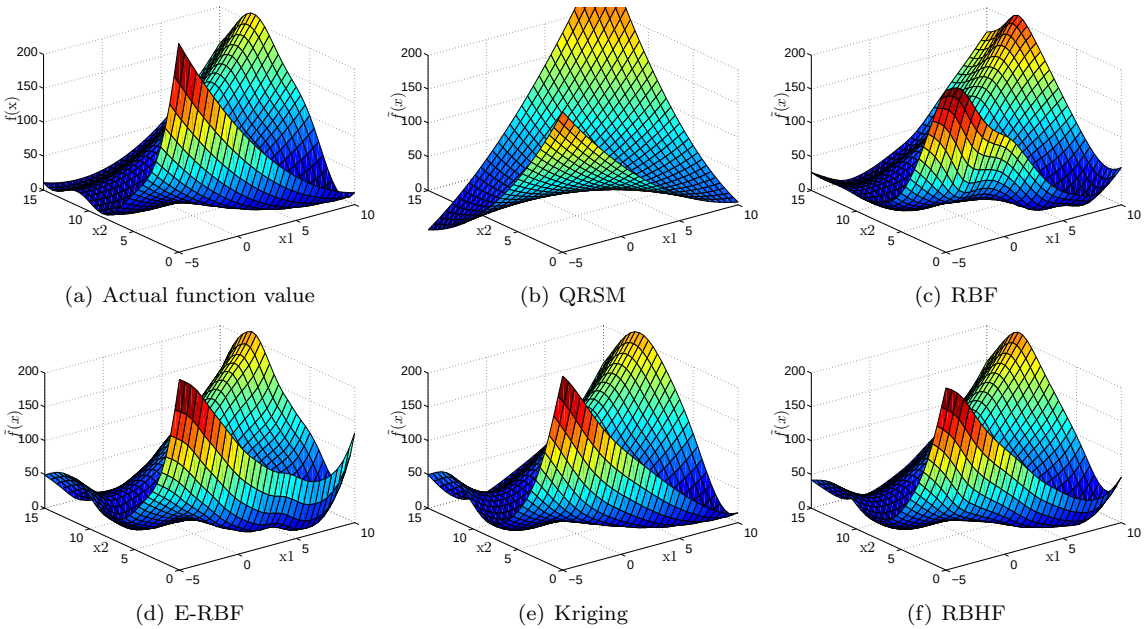
Figure 8. Test function 2 (49 training points)



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

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

Figure 9. Test function 3: Goldstein & Price Function (36 training points)



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

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

Figure 10. Test function 4: Branin-Hoo function (36 training points)

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 RBHF in the 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 reliability of the RBHF surrogate model, are tested on a series of standard examples. The preliminary results show that the RBHF surrogate can provide high fidelity approximations to complex and expensive functional relationships.

In the future, other error metrics that might better represent the performance in the entire design domain should be investigated. The application of the RBHF to a more comprehensive set of test problems should establish the true potential of this method.

VIII. Acknowledgements

Support from the National Found from Awards CMMI-0533330, and CMII-0946765 is gratefully acknowledged.

References

[1]Myers, R. and Montgomery, D., *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, Wiley-Interscience; 2 edition, 2002.

[2]Giunta, A. and Watson, L., "A Comparison of Approximation Modeling Techniques: Polynomial Versus Interpolating Models," Tech. Rep. AIAA-98-4758, 1998.

[3]Sakata, S., Ashida, F., and Zako, M., "Structural Optimization Using Kriging Approximation," *Computer Methods in Applied Mechanics and Engineering*, Vol. 192, No. 7-8, 2003, pp. 923–939.

[4]Simpson, T., *A Concept Exploration Method for Product Family Design*, Ph.D. thesis, Georgia Institute of Technology, 1998.

[5]Cressie, N., *Statistics for Spatial Data*, Wiley, New York, 1993.

[6]Hardy, R. L., "Multiquadric Equations of Topography and Other Irregular Surfaces," *Journal of Geophysical Research*, Vol. 76, 1971, pp. 1905–1915.

[7]Jin, R., Chen, W., and Simpson, T., "Comparative Studies of Metamodelling Techniques Under Multiple Modelling Criteria," *Structural and Multidisciplinary Optimization*, Vol. 23, No. 1, 2001, pp. 1–13.

[8]Cherrie, J. B., Beatson, R. K., and Newsam, G. N., "Fast Evaluation of Radial Basis Functions: Methods for Generalized Multiquadrics in R^n," *SIAM Journal of Scientific Computing*, Vol. 23, No. 5, 2002, pp. 1549–1571.

[9]Hussain, M. F., Barton, R. R., and Joshi, S. B., "Metamodeling: Radial Basis Functions, Versus Polynomials," *European Journal of Operational Research*, Vol. 138, No. 1, 2002, pp. 142–154.

[10]Mullur, A. and Messac, A., "Extended Radial Basis Functions: More Flexible and Effective Metamodeling," *AIAA Journal*, Vol. 43, No. 6, 2005, pp. 1306–1315.

[11]Mullur, A. and Messac, A., "Metamodeling Using Extended Radial Basis Functions: A Comparative Approach," *Engineering with Computers*, Vol. 21, No. 3, 2006, pp. 203–217.

[12]Zhang, J., Chowdhury, S., Messac, A., Castillo, L., and Lebron, J., "Response Surface Based Cost Model for Onshore Wind Farms Using Extended Radial Basis Functions," *ASME 2010 International Design Engineering Technical Conferences (IDETC)*, Montreal, Canada, August 15-18 2010.

[13]Zhang, J., Chowdhury, S., Messac, A., and Castillo, L., "Economic Evaluation of Wind Farms Based on Cost of Energy Optimization," *13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, Fort Worth, Texas, September 13-15 2010.

[14]Zhang, J. Q., Messac, A., Zhang, J., and Chowdhury, S., "Comparison of Surrogate Models Used for Adaptive Optimal Control of Active Thermoelectric Windows," *13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, Fort Worth, Texas, September 13-15 2010.

[15]Duda, R., Hart, P., and Stork, D., *Pattern Classification*, Wiley-Interscience, 2nd ed., 2000.

[16]Yegnanarayana, B., *Artificial Neural Networks*, PHI Learning Pvt. Ltd., 2004.

[17]Clarke, S., Griebsch, J., and Simpson, T., "Analysis of Support Vector Regression for Approximation of Complex Engineering Analyses," *Journal of Mechanical Design*, Vol. 127, No. 6, 2005, pp. 1077–1087.

[18]Vapnik, V., *The Nature of Statistical Learning Theory*, Springer, New York, 1995.

[19]Basudhar, A. and Missoum, S., "Adaptive Explicit Decision Functions for Probabilistic Design and Optimization Using Support Vector Machines," *Computers and Structures*, Vol. 86, No. 19-20, 2008, pp. 1904–1917.

[20]Forrester, A. and Keane, A., "Recent Advances in Surrogate-based Optimization," *Progress in Aerospace Sciences*, Vol. 45, No. 1-3, 2009, pp. 50–79.

[21]Queipo, N., Haftka, R., Shyy, W., Goel, T., Vaidyanathan, R., and Tucker, P., "Surrogate-based Analysis and Optimization," *Progress in Aerospace Sciences*, Vol. 41, No. 1, 2005, pp. 1–28.

[22]Wang, G. and Shan, S., "Review of Metamodeling Techniques in Support of Engineering Design Optimization," *Journal of Mechanical Design*, Vol. 129, No. 4, 2007, pp. 370–380.

[23]Simpson, T., Toropov, V., Balabanov, V., , and Viana, F., "Design and Analysis of Computer Experiments in Multidisciplinary Design Optimization: A Review of How Far We Have Come or Not," *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Victoria, Canada, September 10-12 2008.

[24]Zerpa, L., Queipo, N., Pintos, S., and Salager, J., "An optimization methodology of alkalinesurfactantpolymer flooding processes using field scale numerical simulation and multiple surrogates," *Journal of Petroleum Science and Engineering*, Vol. 47, No. 3-4, 2005, pp. 197–208.

[25]Goel, T., Haftka, R., Shyy, W., and Queipo, N., "Ensemble of Surrogates," *Structural and Multidisciplinary Optimization*, Vol. 33, No. 3, 2007, pp. 199–216.

[26]Sanchez, E., Pintos, S., and Queipo, N., "Toward an Optimal Ensemble of Kernel-based Approximations with Engineering Applications," *Structural and Multidisciplinary Optimization*, Vol. 36, No. 3, 2008, pp. 247–261.

[27]Acar, E. and Rais-Rohani, M., "Ensemble of Metamodels with Optimized Weight Factors," *Structural and Multidisciplinary Optimization*, Vol. 37, No. 3, 2009, pp. 279–294.

[28]Viana, F., Haftka, R., and Steffen, V., "Multiple Surrogates: How Cross-validation Errors Can Help Us to Obtain the Best Predictor," *Structural and Multidisciplinary Optimization*, Vol. 39, No. 4, 2009, pp. 439457.

[29]Deb, K., *Multi-Objective Optimization Using Evolutionary Algorithms*, Wiley, 2001.

[30]Forrester, A., Sobester, A., and Keane, A., *Engineering Design via Surrogate Modelling: A Practical Guide*, Wiley, 2008.

[31]McKay, M., Conover, W., and Beckman, R., "A comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, Vol. 21, No. 2, 1979, pp. 239–245.

[32]Lophaven, S., Nielsen, H., and Sondergaard, J., "DACE a matlab kriging toolbox, version 2.0," Tech. Rep. Informatics and mathematical modelling report IMM-REP-2002-12, Technical University of Denmark, 2002.