

# Various approaches for constructing an ensemble of metamodels using local measures

Erdem Acar

Received: 11 October 2009 / Revised: 19 April 2010 / Accepted: 16 May 2010 / Published online: 12 June 2010  
© Springer-Verlag 2010

**Abstract** Metamodels are approximate mathematical models used as surrogates for computationally expensive simulations. Since metamodels are widely used in design space exploration and optimization, there is growing interest in developing techniques to enhance their accuracy. It has been shown that the accuracy of metamodel predictions can be increased by combining individual metamodels in the form of an ensemble. Several efforts were focused on determining the contribution (or weight factor) of a metamodel in the ensemble using *global* error measures. In addition, prediction variance is also used as a *local* error measure to determine the weight factors. This paper investigates the efficiency of using local error measures, and also presents the use of the *pointwise cross validation error* as a local error measure as an alternative to using prediction variance. The effectiveness of ensemble models are tested on several problems with varying dimensionality: five mathematical benchmark problems, two structural mechanics problems and an automobile crash problem. It is found that the spatial ensemble models show better performances than the global ensemble for the low-dimensional problems, while the global ensemble is a more accurate model than the spatial ensembles for the high-dimensional problems. Ensembles based on pointwise cross validation error and prediction variance provide similar accuracy. The ensemble models based on local measures reduce cross validation errors drastically, but their performances are not that impressive in

reducing the error evaluated at random test points, because the pointwise cross validation error is not a good surrogate for the error at a point.

**Keywords** Ensemble · Local error measures · Metamodeling · Surrogate modeling

## Nomenclature

$\mathbf{C}$	mean square error matrix
$e_{ik}$	cross validation error of the $i$ th metamodel at the $k$ th training data point
$E_i$	generalized root mean square cross validation error (GMSE) of the $i$ th metamodel
GLO	the global ensemble model based on the minimization of the global error measure GMSE
GP	Gaussian process
$I_k(x)$	a function that relates the weight factor at the $k$ th training point to the weight factor at any point $x$
KR0, KR1	Kriging metamodels with constant and linear trend models, respectively
$N_M$	number of metamodels used in the ensemble
PRS2	second-order polynomial response surface (with all terms in the polynomial included)
RBF	radial basis functions
SPV	the spatial ensemble model based on minimization of prediction variance
SP1–SP4	four different spatial ensemble models proposed in this paper
$w_i(x)$	normalized weight factor of the $i$ th metamodel at any point $x$ ( $0 \leq w_i(x) \leq 1$ )
$W_i(x)$	unnormalized weight factor of the $i$ th metamodel at any point $x$

E. Acar (✉)  
Department of Mechanical Engineering,  
TOBB University of Economics and Technology,  
Sogutozu Cad. No: 43, Sogutozu,  
Ankara 06560, Turkey  
e-mail: acar@etu.edu.tr

$W_{ik}$	unnormalized weight factor of the $i$ th metamodel at $k$ th data point
$\hat{y}_{ens}$	response prediction of the ensemble of metamodels
$\hat{y}_i$	response prediction of the $i$ th metamodel

## 1 Introduction

The analysis of complex engineering systems relies on high fidelity computer simulations that require several hours of CPU time. When high-fidelity simulations are combined with numerical design optimization, the computational cost becomes excessive. Hence, metamodels (or surrogate models) are widely used to replace the computationally expensive simulations with smooth analytic functions for efficient estimation of system responses.

Metamodeling techniques aim at regression and/or interpolation fitting of the system responses at some specified training (observation) points. There are many metamodeling techniques including polynomial response surface approximations, PRS (Box et al. 1978; Myers and Montgomery 2002), radial basis functions, RBF (Dyn et al. 1986; Mullur and Messac 2004), Kriging, KR (Sacks et al. 1989; Martin and Simpson 2005), and Gaussian process, GP (MacKay 1998; Rasmussen and Williams 2006).

A review of literature provides many examples where the accuracy and the efficiency of various metamodels for linear, nonlinear, smooth, and noisy responses have been investigated (Giunta and Watson 1998; Simpson et al. 2001; Jin et al. 2001; Stander et al. 2004; Fang et al. 2005; Wang et al. 2006). For instance, Fang et al. (2005) found RBF gives accurate metamodels for highly nonlinear responses, Simpson et al. (2001) found Kriging to be most suitable for slightly nonlinear responses in high dimension spaces, and Jin et al. (2001) proposed the use of PRS for slightly nonlinear and noisy responses. An extensive review of metamodeling can be found in Queipo et al. (2005) and Wang and Shan (2007).

While the studies cited above identify a single metamodel as being accurate for a particular form of response, it is not always apparent for an engineer to know in prior which metamodel is the best for a specific application. Therefore, as an alternative to using a single metamodel, several researchers combined multiple metamodels in the form of an ensemble (Goel et al. 2007; Sanchez et al. 2008; Acar and Rais-Rohani 2009; Viana et al. 2009). These studies showed that the resulting ensemble of metamodels takes advantage of the prediction ability of each individual metamodel to enhance the accuracy of the response predictions.

Goel et al. (2007), Acar and Rais-Rohani (2009) and Viana et al. (2009) used the generalized mean square cross validation error (GMSE) as a global metric to determine the

contribution (or weight factor) of an individual metamodel in the ensemble. Since a global error metric was used, the weight factors were kept constant over the entire design space. Instead of using a global error measure, Sanchez et al. (2008) proposed the use of the prediction variance as a local error measure to determine the weight factors of the metamodels in the ensemble. Since Sanchez et al. (2008) used a local error metric, the weight factors were flexible over the design space leading to better predictions. On the other hand, Viana et al. (2009) showed that the cross-validation errors are better indicators of the global accuracy of a metamodel than prediction variance. Motivated from this finding, this paper proposes the use of the pointwise cross validation error as a local error measure as an alternative to using prediction variance. The main objective of the paper is to investigate the efficiency of using various local error measures.

The remainder of the paper is structured as follows. Section 2 presents the basic ensemble formulation and weight factor selection using global error measures. Section 3 proposes different approaches for selecting weight factors using local error measures. Section 4 describes the example problems considered, and Section 5 describes the numerical procedure followed while constructing the ensembles. The results are presented and discussed in Section 6, and a summary of important conclusions is listed in Section 7.

## 2 Ensemble of metamodels constructed using global error measures

Traditional application of metamodeling techniques is based on constructing many different metamodels, selecting the best one and discarding the rest. This practice has two major shortcomings. First, most of the effort spent on constructing different metamodels is wasted. Second, the performances of different metamodels are dependent on the training data set used, so it cannot be guaranteed that the selected metamodel will perform the best as a new data is available. These drawbacks can be overcome by using an ensemble of metamodels rather than a single one. This section provides a brief description of ensemble of metamodels.

An ensemble can be constructed by using a weighted average of different metamodels. The resulting ensemble model can be defined as

$$\hat{y}_{ens}(x) = \sum_{i=1}^{N_M} w_i(x) \hat{y}_i(x) \quad (1)$$

where  $\hat{y}_{ens}$  is the prediction of the ensemble,  $N_M$  is the number of metamodels used,  $w_i$  is the weight factor for the  $i$ th

metamodel and  $\hat{y}_i$  is the prediction of the  $i$ th metamodel. The weight factors satisfy

$$\sum_{i=1}^{N_M} w_i(x) = 1 \tag{2}$$

Construction of an accurate ensemble requires judicious selection of the weight factors. The weight factors,  $w_i$ , for the metamodels need to be selected such that the prediction accuracy of the ensemble is maximized. Different weight selection procedures followed by different researchers are briefly discussed in the followings.

Goel et al. (2007) considered an ensemble of three metamodels (PRS, KR, and RBF) and used the generalized mean square cross validation error (GMSE) as a global error metric to select the weight factors using a heuristic formulation:

$$w_i = \frac{w_i^*}{\sum_{j=1}^M w_j^*} \tag{3.1}$$

$$w_i^* = (E_i + \alpha \bar{E})^\beta \tag{3.2}$$

$$\bar{E} = \frac{1}{N_M} \sum_{i=1}^{N_M} E_i \tag{3.3}$$

where  $E_i$  is the GMSE of the  $i$ th metamodel calculated from

$$E_i = \frac{1}{N} \sum_{k=1}^N (y^{(k)} - \hat{y}_i^{(k)})^2 \tag{4}$$

where  $y^{(k)}$  is the true response at the  $k$ th data point  $\mathbf{x}_k$ , and  $\hat{y}_i^{(k)}$  is the corresponding predicted value from the metamodel constructed using all except the  $k$ th data point. The parameters  $\beta < 0$  and  $\alpha < 1$  are selected by the analyst based on the importance of  $E_i$  and  $\bar{E}$ . Goel et al. (2007) found  $\alpha = 0.05$  and  $\beta = -1$  leads to a good model in their study. Even though the analyst has the flexibility on the values of these parameters, the optimal values of these parameters may not be known beforehand. Notice that in this model the weight factors of metamodels in the ensemble are constant over the entire design space.

Acar and Rais-Rohani (2009) also used the GMSE as the global error metric of interest and proposed that the weight factors of different metamodels can be selected via solving the following optimization problem

Find  $w_i$  (5.1)

Minimize  $GMSE[\hat{y}_{ens}(w_i)]$  (5.2)

Such that  $\sum_{i=1}^{N_M} w_i = 1$  (5.3)

Notice that in this model the weight factors do not depend on the pointwise location of the prediction point.

Similarly, Viana et al. (2009) proposed to select the weight factors following an approach based on minimizing the mean square error (MSE). They found that the optimum values of weight factors can be calculated from

$$\mathbf{w} = \frac{\mathbf{C}^{-1}\mathbf{1}}{\mathbf{1}^T\mathbf{C}^{-1}\mathbf{1}} \tag{6}$$

where  $\mathbf{1}$  is the identity matrix, and the elements of the  $\mathbf{C}$  matrix (the mean square error matrix) can be calculated from

$$C_{ij} = \frac{1}{N} E_i^T E_j \tag{7}$$

where  $N$  is the number of data points, and  $E_i$  is the GMSE of the  $i$ th metamodel as defined in (4). Notice that the ensemble model proposed by Viana et al. (2009) specifies also constant weight factors for metamodels over the entire design space. Note also that boldface symbols appearing in the equations define matrices.

### 3 Constructing ensembles using local error measures

Instead of using a global error metric leading to constant weight factors over the entire design space, the use of a local error measures may lead to more accurate predictions by allowing flexible weight factors over the design space. Sanchez et al. (2008) proposed the use of the prediction variance as the local error metric, and set the value of weight factor for each metamodel to be inversely proportional to the pointwise estimate of the prediction variance as

$$w_i(x) = \frac{\frac{1}{V_i(x)}}{\sum_{j=1}^{N_M} \frac{1}{V_j(x)}} \tag{8}$$

where  $V_i(x)$  is the pointwise prediction variance of the  $i$ th metamodel. Here, the prediction variances are calculated using the empirical formula proposed by Sanchez et al. (2008) using the  $k$  nearest neighbors of point  $x$ . That is, the prediction variance of the  $j$ th metamodel is computed from

$$V_j(x) = \frac{1}{k-1} \sum_{n=1}^k [y(x_n) - \hat{y}_j(x_n)]^2 \tag{9}$$

where  $x_1, x_2, \dots, x_k$  are the  $k$  nearest neighbor data points of the prediction point  $x$ . Here,  $k=3$  is used as proposed by Sanchez et al. (2008). Notice from (8) that the weight factors in the ensemble proposed by Sanchez et al. (2008)

depend on the pointwise location of the prediction point  $x$ . However, setting the values of weight factors to be inversely proportional to the prediction variances of metamodells is an optimal selection only if the metamodell predictions are unbiased and uncorrelated, which is not always the case. In addition, it has been shown that the cross-validation errors are better indicators of the global accuracy of a metamodell than prediction variance (Viana et al. 2009). Therefore, the following approaches, which use the pointwise cross validation error as the local error measure of interest, are proposed in this paper. The main objective of the paper is to investigate the efficiency of the use of various local error measures.

### Proposed approaches

In this paper, the weight factors are selected based on cross-validation error rather than the prediction variance. Here, the weight factors are proposed to be calculated from

$$w_i(x) = \frac{W_i(x)}{\sum_{j=1}^{N_M} W_j(x)} \quad (10)$$

where  $W_i(x)$  is the weight factor of the  $i$ th metamodell. Recall that we need weight factors at prediction points, but the cross-validation errors are defined at the data points. To solve this problem, this paper proposes that first the weight factors at data points are calculated, and then the weight factors at the prediction points are computed from those. That is, the weight factors at a prediction point depend on two factors: (a) the distance between the prediction point and the data points, (b) the weight factors at data points.

This paper envisions two different ways to assess the weight factors at prediction points from the weight factors at data points:

- (i) The weight factors at data points are scaled inversely proportional to the square of the distance between the prediction point and the data point.
- (ii) The weight factors at a prediction point are equal to the weight factors at the closest data point.

Similarly, two different ways are proposed to compute the weight factors at data points

- (i) The weight factors at data points are selected such that weight factor of individual metamodell with smallest cross validation error is one, while all other metamodells have zero weights.
- (ii) The weight factors at data points are inversely proportional to the square of the cross-validation errors.

Therefore, overall four different heuristic approaches are proposed in this paper to compute the weight factors  $W_i(x)$  as described below. Since the weight factors  $W_i(x)$  are dependent on the spatial location of the prediction point  $x$ , these ensemble of models can be called as spatial ensemble of metamodells.

*Approach 1* To compute the weight factors at a prediction point, the weight factors at data points are scaled inversely proportional to the square of the distance between the prediction point and the data point. The weight factors at data points are selected such that weight factor of individual metamodell with the smallest cross validation error is one, while the other metamodells have zero weights. Hence, the spatial weight factors  $W_i(x)$  in (10) are computed from

$$W_i(x) = \sum_{k=1}^N W_{ik} I_k(x) \quad (11)$$

where  $W_{ik}$  is the pointwise weight factor of the  $i$ th metamodell at  $k$ th data point.  $W_{ik}$  is equal to one for the metamodell with the smallest cross validation error at the  $k$ th data point, and equal to zero for all other metamodells at the  $k$ th data point. That is, at the  $k$ th data point, only the prediction of the most accurate individual metamodell is utilized.

The function  $I_k(x)$  in (11) is computed from

$$I_k(x) = \frac{1}{d_k^2(x)} \quad (12)$$

where the distance between the prediction point  $x$  and the data point  $x_k$  is calculated from

$$d_k(x) = \|x - x_k\| \quad (13)$$

where  $\| \cdot \|$  is the Euclidian norm. In (12), if  $d_k(x)$  is computed as zero, there will be a singularity. To solve this problem,  $W_i(x) = W_{ik}$  is used when  $d_k(x)=0$ .

*Approach 2* To compute the weight factors at a prediction point, the weight factors at data points are scaled inversely proportional to the square of the distance between the prediction point and the data point. The weight factors at data points are inversely proportional to the square of the cross-validation errors. Hence, the spatial weight factors are computed from (11), where the pointwise weight factors are calculated from

$$W_{ik} = \frac{1}{e_{ik}^2}, \quad e_{ik} = y_{(k)} - \hat{y}_{i(k)} \quad (14)$$

Here  $y_{(k)}$  is the true response at the data point  $x_k$ , and  $\hat{y}_{i(k)}$  is the corresponding predicted value of the  $i$ th metamodell constructed using all except the  $k$ th data point. In this approach, the function  $I_k(x)$  is computed from (12) and (13).

**Approach 3** The weight factors at a prediction point are equal to the weight factors at the closest data point. The weight factors at data points are selected such that weight factor of individual metamodel with smallest cross validation error is one, while all other metamodels have zero weights. The spatial weight factors are computed from (11). The pointwise weight factor  $W_{ik}$  is equal to one if a metamodel has the smallest cross validation error at  $k$ th data point, and equal to zero otherwise. Similarly, the function  $I_k(x)$  is equal to one if a data point  $x_k$  is closest to the prediction point  $x$ , and equal to zero otherwise. In short, at a prediction point, first the closest data point is found, and then the metamodel with the minimum cross-validation point at that data point is used for ensemble prediction.

**Approach 4** The weight factors at a prediction point are equal to the weight factors at the closest data point. The weight factors at the data points are inversely proportional to the square of the cross validation errors. That is, the spatial weight factors are computed from (11), and the pointwise weight factor  $W_{ik}$  is computed from (14).  $I_k(x)$  is equal to one if a data point  $x_k$  is closest to the prediction point  $x$ , and equal to zero otherwise.

### 4 Example problems

The performances of five individual metamodels, the global ensemble model of Acar and Rais-Rohani (2009), the spatial ensemble model of Sanchez et al. (2008) and the proposed spatial models are evaluated considering eight example problems. The first five examples are widely used mathematical benchmark problems in the literature. The following two examples are structural mechanics problems, where the responses are described by analytic functions. In the last example problem, an automobile crash problem, the critical responses are obtained from finite element simulations.

#### 4.1 Mathematical problems

The mathematical benchmark problems are defined by the following analytical functions:

Branin-Hoo function (two-variable)

$$y(x_1, x_2) = \left(x_2 - \frac{5.1x_1^2}{4\pi^2} + \frac{5x_1}{\pi} - 6\right)^2 + 10 \left(1 - \frac{1}{8\pi}\right) \cos(x_1) + 10 \tag{15}$$

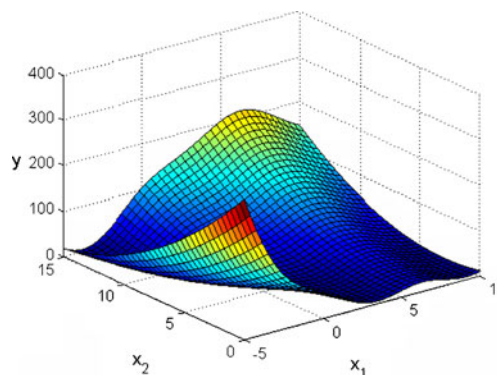


Fig. 1 Branin-Hoo function

where  $x_1 \in [-5, 10]$ , and  $x_2 \in [0, 15]$ . Branin-Hoo function in the specified range is depicted in Fig. 1.

Goldstein-Price function (two-variable)

$$y(x_1, x_2) = \left[1 + (x_1 + x_2 + 1)^2 \times (19 - 14x_1 + 13x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)\right] \times \left[30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)\right] \tag{16}$$

where  $x_1, x_2 \in [-2, 2]$ . Goldstein-Price function in the specified range is depicted in Fig. 2.

Hartman function (six-variable)

$$y(\mathbf{x}) = -\sum_{i=1}^m c_i \exp\left[-\sum_{j=1}^n a_{ij}(x_j - p_{ij})^2\right] \tag{17}$$

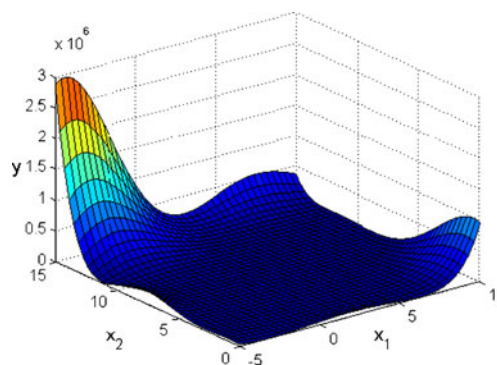


Fig. 2 Goldstein-Price function

**Table 1** Parameters used in six-variable Hartman function,  $j = 1, \dots, 6$

$i$	$a_{ij}$					$c_i \quad p_{ij}$								
1	10.0	3.0	17.0	3.5	1.7	8.0	1.0	0.1312	0.1696	0.5569	0.0124	0.8283	0.5886	
2	0.05	10.0	17.0	0.1	8.0	14.0	1.2	0.2329	0.4135	0.8307	0.3736	0.1004	0.9991	
3	3.0	3.5	1.7	10.0	17.0	8.0	3.0	0.2348	0.1451	0.3522	0.2883	0.3047	0.6650	
4	17.0	8.0	0.05	10.0	0.1	14.0	3.2	0.4047	0.8828	0.8732	0.5743	0.1091	0.0381	

where  $x_1 \in [0, 1]$ . Here six-variable ( $n = 6$ ) model of this function is considered, where  $m$  is taken four. The values of function parameters  $c_i, a_{ij}$  and  $p_{ij}$ , taken from Goel et al. (2007), are provided in Table 1.

Extended Rosenbrock function (nine-variable)

$$y(\mathbf{x}) = \sum_{i=1}^{m-1} \left[ (1 - x_i)^2 + 100 (x_{i+1} - x_i^2)^2 \right] \quad (18)$$

where  $x_1 \in [-5, 10]$ . Here nine-variable ( $m = 9$ ) model of this function is considered.

Dixon-Price function (12-variable)

$$y(\mathbf{x}) = \sum_{i=1}^{m-1} \left[ (1 - x_i)^2 + 100 (x_{i+1} - x_i^2)^2 \right] \quad (19)$$

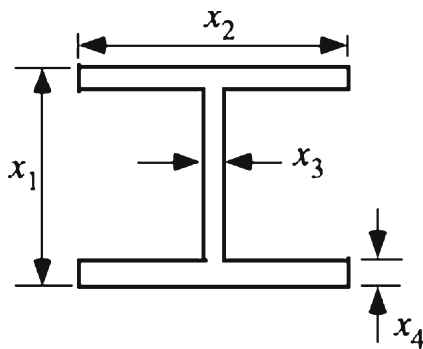
where  $x_1 \in [-5, 10]$ . Here 12-variable ( $m = 12$ ) model of this function is considered.

## 4.2 Structural mechanics problems

### 4.2.1 Four variable I-beam

This four-variable I-beam (see Fig. 3) problem is taken from Messac and Mullur (2008). The critical response for this problem is the maximum bending stress developed in the beam, which is calculated from

$$\sigma_{\max} = \frac{P \cdot x_1}{I}, \quad I = \frac{1}{12} \left[ x_2 x_1^3 - (x_2 - x_3) (x_1 - 2x_4)^3 \right] \quad (20)$$



**Fig. 3** The cross-section of the four variable I-beam design

The ranges of the design variables are taken as  $0.1 m \leq x_1, x_2 \leq 0.8 m$  and  $0.009 m \leq x_3, x_4 \leq 0.05 m$  as specified in Messac and Mullur (2008).

### 4.2.2 Fortini's clutch

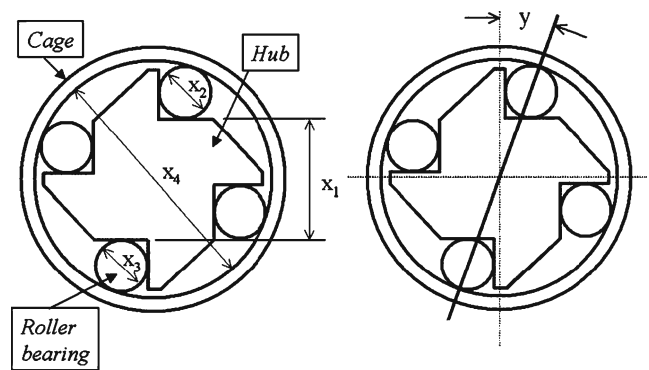
The other four-variable problem is taken from Lee and Kwak (2006). This overrunning clutch assembly, depicted in Fig. 4, is known as Fortini's clutch. The contact angle  $y$  is given in terms of the geometric variables  $x_1$  through  $x_4$  as

$$y = \arccos \left[ \frac{x_1 + 0.5 (x_2 + x_3)}{x_4 - 0.5 (x_2 + x_3)} \right] \quad (21)$$

The problem specified in Lee and Kwak (2006) is a reliability assessment problem for the clutch. The mean and standard deviations of the geometric variables are provided in Table 2. The ranges for these variables are taken as  $\pm$  five times standard deviations away from the mean values.

## 4.3 Automobile crash problem

A finite element (FE) model of a c-class passenger car shown in Fig. 5 is used for offset-frontal impact (OFI) simulations using the FE code, LS-DYNA. In this example, metamodels are constructed to estimate the intrusion distances and average peak accelerations at the floor pan in OFI scenario for crash duration of 100 ms. There are two responses of interest (intrusion distance and acceleration) for this example.



**Fig. 4** The clutch assembly (courtesy of Lee and Kwak 2006)

**Table 2** Mean and standard deviation of the geometric variables

Variable	Mean	Standard deviation
$x_1$	55.29	0.0793
$x_2$	22.86	0.0043
$x_3$	22.86	0.0043
$x_4$	101.60	0.0793

The input-variable vector consists of the geometric parameters that control the shape ( $x_1$  to  $x_4$ ) and wall thickness ( $x_5$ ) of the two side rails (see Fig. 5) as well as the parameters that define variability or uncertainty in the material stress–strain relationship ( $x_6$ ), offset distance ( $x_7$ ), impact speed ( $x_8$ ), and occupant mass ( $x_9$ ). The input variables are normalized between  $-1$  and  $+1$  in this example. A more detailed description of this problem can be found in Rais-Rohani et al. (2006).

## 5 Numerical procedure

### 5.1 Design and analysis of computer experiments

For the mathematical problems in (15)–(19) and the structural mechanics problems in (20), (21), Latin hypercube sampling (LHS) technique is used to select the locations of the training points such that the minimum distance between the design points is maximized. The MATLAB® routine “lhsdesign” and “maximin” criterion with a maximum of 100 iterations is used to obtain the locations of the training points. Random sampling is used to generate 1,000 test points for a specified training set. It should be noted that for 12-variable problems, 1,000 test points in a single test set is relatively small. However, since 1,000 different test sets are used, the mean values of the errors calculated over 1,000 test sets provides a general trend for the errors.

To reduce the effect of random sampling, a varying number of different training sets are used for the mathematical examples and the structural mechanics examples (see Table 3, column 2). The low computational cost allowed considering repetitive training and test sets. To keep the computational cost affordable, the number of training sets is reduced as the number of variables is increased. The number of the training points in a set is selected as twice the number of coefficients in a full quadratic PRS. Hence, all the metamodels are constructed multiple times with the error estimate being the average value corresponding to multiple versions (replicates) of the same metamodel. In addition, for each training set, a different set of test points is used to reduce the bias in RMSE estimation. The accuracy of metamodels for the benchmark problems is measured using the mean and the coefficient of variation (COV) of GMSE and RMSE error metrics.

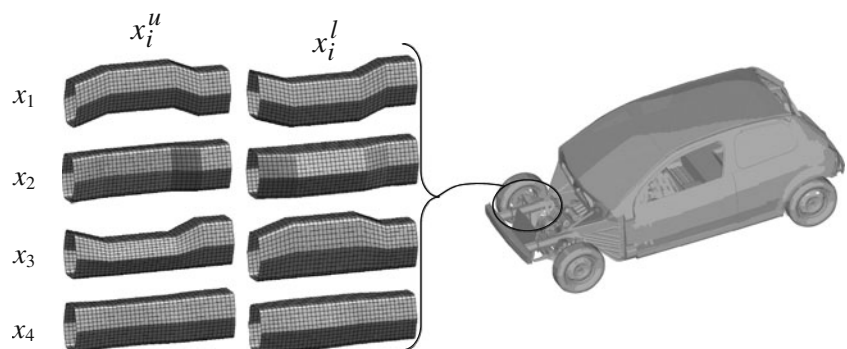
For the automobile crash problem, the LHS technique is also used; however, because of the large computational cost of each high-fidelity simulation (13 CPU hours for FFI and 17 CPU hours for OFI using a 32-processor IBM Linux Cluster with Intel Pentium III 1.266 GHz processors and 607.5 GB RAM), the metamodeling calculations are done using only a single training set with 100 training points. A summary of the training and test data sets used in each problem is provided in Table 3.

To compute the weight factors, the “fmincon” function (optimizer) of MATLAB based on the sequential quadratic programming algorithm is used to solve the optimization problem. Since the optimization routine is a gradient-based optimizer and the objective function being minimized is not necessarily convex, a multiple starting point strategy is used to increase the probability for the solution to converge to a global optimum.

### 5.2 Metamodeling techniques

As noted earlier, four different metamodeling techniques are considered in this study: PRS, RBF, KR, and GP. In the

**Fig. 5** Perturbed geometry of the right side rail at the upper and lower limits of  $x_1$  through  $x_4$



**Table 3** Summary of training and test data used in each problem

Problem	Training sets	Number of variables	Design points in a training set	Test points in a set
Branin-Hoo	1,000	2	12, 24, 36	1,000
Goldstein-Price	1,000	2	12	1,000
I-beam design	1,000	4	30	1,000
Fortini's clutch	1,000	4	30	1,000
Hartman	400	6	56	1,000
Extended Rosenbrock	100 and 20	9	110 and 220	1,000
Dixon-Price	20	12	182	1,000
Automobile crash	1	9	100	40

case of Kriging, both constant trend model, KR0, and linear trend model KR1 are used. Therefore, the ensemble models are composed of five members.

The PRS metamodel is represented by fully quadratic polynomial. The RBF metamodel is based on the multi-quadratic formulation with the constant,  $c = 1$ . In Kriging metamodels KR0 and KR1, a Gaussian correlation function is used. The covariance function in GP metamodel is selected as the sum of a squared exponential function with automatic relevance determination and covariance function for the input-independent noise (i.e., white noise). The mathematical descriptions of these metamodels are provided in the [Appendix](#).

## 6 Results and discussion

In this section root mean square error (RMSE) values evaluated at test points are reported. Since the weight factor selection of the global ensemble model and the proposed spatial ensemble models used GMSE metric, the RMSE is the main error metric of interest in this paper. To explore the effect of dimensionality, example problems with varying dimension (from two to 12) are considered. Moreover, to see the effects of the number of training points, the Branin-Hoo (a 2-D function) and the extended Rosenbrock (a 9-D function) functions are evaluated with varying number of training points. With the exception of the automobile

crash example, the mean and coefficient of variation (COV) values of RMSE over various training and test sets are computed. The error values are normalized with respect to the most accurate individual metamodel to provide a better comparison of different models.

The abbreviated symbols introduced earlier are used to identify the individual metamodels. The ensemble model of Acar and Rais-Rohani (2009) based on the minimization of the global error measure GMSE is denoted by GLO. The spatial ensemble of Sanchez et al. (2008) that uses prediction variance is labeled by SPV. Finally, the spatial ensembles proposed in Section 3 of this paper are denoted by SP1 through SP4.

### 6.1 Branin-Hoo function (two-variable)

The comparison of the GMSE of all individual metamodels as well as the ensemble models for the Branin-Hoo function is provided in Table 4. At first, 12 points are used in a training set, and later on the number of points is increased to 24 and 36 to explore the effects of the number of training points. The smallest error value in each category is shown in bold for ease of comparison. According to the average value of GMSE, RBF is found to be the most accurate individual metamodel for this function. All the ensemble models are found to yield smaller GMSE than RBF. When the ensemble models are compared, the spatial ensemble based on the first and third approaches (SP1 and SP3) yield the smallest

**Table 4** Comparison of normalized GMSE of individual and ensemble models for the Branin-Hoo function

	Individual metamodels					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
Mean	1.24	<b>1.00</b>	1.03	1.30	1.05	0.82	0.85	<b>0.65</b>	0.78	<b>0.65</b>	0.78
COV	0.39	0.31	0.29	0.31	0.30	0.35	0.36	0.43	0.41	0.43	0.41

The smallest error value in each category is shown in bold for ease of comparison. The number of points in a training set is 12



**Table 5** Comparison of normalized RMSE of individual and ensemble models for the Branin-Hoo function

	Individual metamodels					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
Mean	1.26	<b>1.00</b>	1.01	1.21	1.03	0.99	<b>0.96</b>	0.97	<b>0.96</b>	1.05	0.99
COV	0.13	0.21	0.36	0.39	0.42	0.27	0.26	0.26	0.27	0.24	0.25

The smallest error value in each category is shown in bold for ease of comparison. The number of points in a training set is 12

GMSE overall. With an average GMSE of 0.65, SP1 and SP3 are 35% more accurate than RBF and about 20% more accurate than the global ensemble. The GMSE of the spatial ensemble based on prediction variance SPV is larger than other spatial ensembles, since GMSE and prediction variance are not closely correlated.

Table 4 shows that the spatial ensembles SP1 and SP3 have the same performance. This is not surprising since the GMSE error metric is evaluated at the data points, and the weight factor selection methods of these approaches at the data points are essentially the same. Similarly, the spatial ensembles SP2 and SP4 have the same performance due to the same reason.

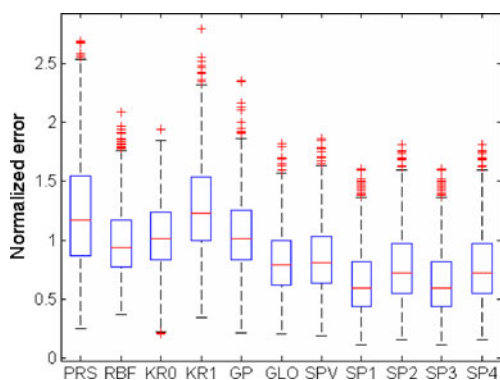
Recall that the mean and COV values in Table 4 are calculated based on 1,000 different training sets. Hence, the mean values of the GMSE over the selected population sample has a COV of  $1/\sqrt{1,000}$  times that of the native COV of the GMSE. For instance, the COV of the mean GMSE for PRS model is  $0.39/\sqrt{1,000} = 0.012$ . This number provides an estimate of the standard error in the prediction of mean GMSE over 1,000 training sets, which is fairly small in this case. For instance, if 100 different training sets were used instead, then the standard error in the mean GMSE prediction would be around 4%, which would be quite large.

Instead of the GMSE evaluated at the data points, the RMSE evaluated at random test points may provide a better

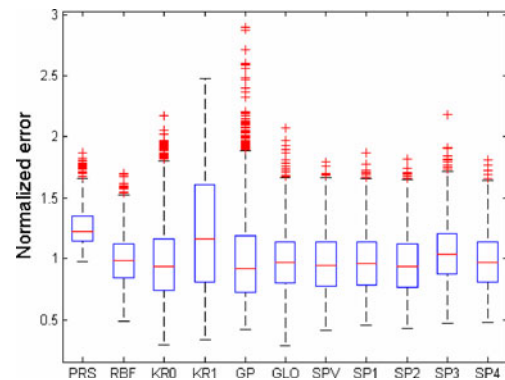
measure of the accuracy of metamodels. The comparison of the RMSE of all models for the Branin-Hoo function is given in Table 5. Comparison of the results given in Tables 4 and 5 reveals that GMSE and RMSE of the individual metamodels are similar in terms of both magnitude and trend. However, GMSE and RMSE behaviors of the ensemble models are different. Even though the ensemble models, in particular SP1 and SP3 models, show excellent performances if measured by GMSE, their performances are not that impressive if measured by RMSE. In addition, it is found that the accuracies of the spatial ensembles that use pointwise cross validation and the accuracy of the spatial ensemble that uses prediction variance are close.

Table 5 shows that the RBF is the most accurate individual metamodel for this function. Table 5 also shows that all ensemble models, except SP3, provide more accurate predictions than the best individual metamodel RBF. Amongst all the ensemble models, the spatial ensembles SPV and SP2 are the most accurate. The COV values provided in the last row indicate that the standard errors in the prediction of mean RMSEs over 1,000 training sets are fairly small as in the case of GMSE. Therefore, for the remaining example problems, only the mean values of RMSE and GMSE of the metamodels will be given.

Figures 6 and 7 show the boxplots for the error metrics, GMSE and RMSE, respectively, corresponding to the 11



**Fig. 6** Boxplots of normalized GMSE over 1,000 training sets for the Branin-Hoo function. The number of points in a training set is 12



**Fig. 7** Boxplots of normalized RMSE over 1,000 training and test sets for the Branin-Hoo function. The number of points in a training set is 12

**Table 6** Comparison of normalized RMSE of individual and ensemble models for the Branin-Hoo function when the number of points in a training set is increased

	Individual metamodells					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
Number of points in training set is increased to 24											
GMSE <sup>a</sup>	4.75	3.26	<b>1.00</b>	1.13	1.47	0.85	0.91	<b>0.68</b>	0.84	<b>0.68</b>	0.84
RMSE <sup>a</sup>	6.92	4.00	<b>1.00</b>	1.07	1.30	<b>0.95</b>	1.08	1.23	1.23	1.69	1.52
Number of points in training set is increased to 36											
GMSE <sup>a</sup>	59.7	31.9	3.90	4.10	<b>1.00</b>	0.79	0.80	0.68	0.75	0.68	0.75
RMSE <sup>a</sup>	179	77.8	7.66	7.97	<b>1.00</b>	<b>1.40</b>	1.98	2.46	2.86	3.90	4.30

The smallest error value in each category is shown in bold for ease of comparison

<sup>a</sup>The mean values of these error metrics over 1,000 training and test sets

different metamodells for the Branin-Hoo function. The box-plots provide a graphical depiction of how the normalized value of each metric varies over the range of training sets used. The bottom and top of each box represent the lower and upper quartile values, respectively, with the interior line representing the median. The broken line (whiskers) extending from each end of the box indicates the extent of the remaining data relative to the lower and upper quartiles. Here, the maximum whisker length is set at 1.5 times the inter-quartile range, and the data beyond this limit (if present) are characterized as outliers and represented by the + symbols.

Finally, for the Branin-Hoo function (chosen as the representative example for low-dimensional problems), the effect of increasing the number of training points is explored. The number of training points is increased from 12 to 24 and 36, and the overall analysis is repeated. The results are provided in Table 6. Combining the results presented in Tables 5 and 6, it is seen that as the number of training points is increased (1) the order of the accuracies of the metamodells changes, (2) the deviation between the accuracies of the metamodells increases, (3) the global ensemble becomes more accurate than the spatial ensembles, and (4) the best individual metamodell becomes more accurate than all ensemble models.

## 6.2 Goldstein-Price function (two-variable)

For the other two-variable example problem, Goldstein-Price function, the values of GMSE and RMSE for the individual and the ensemble of metamodells are listed in Table 7. For the Goldstein-Price function, the RBF is found to be the most accurate individual metamodell whether evaluated by GMSE or RMSE. Amongst the ensemble models, the spatial ensembles SP1 and SP3 are found to be the most accurate if evaluated by the GMSE. The spatial ensemble SP1 is found to be the most accurate if evaluated by the RMSE. Notice that the best ensemble model is better than the best individual metamodell if GMSE is used, while this is not the case if RMSE is used.

## 6.3 Four variable structural mechanics problems

The results for the two structural mechanics problems are given in Tables 8 and 9. When GMSE is the error metric of interest, RBF is found to be the most accurate individual metamodell and the spatial ensembles SP1 and SP3 are found to be the most accurate ensemble models for the four-variable I-beam problem (see Table 8). For the I-beam problem, the best ensembles SPV and SP1 are also found

**Table 7** Comparison of accuracies of individual and ensemble models for the Goldstein-Price function

	Individual metamodells					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
GMSE <sup>a</sup>	1.09	<b>1.00</b>	1.11	1.19	1.15	0.91	0.96	<b>0.80</b>	0.91	<b>0.80</b>	0.91
RMSE <sup>a</sup>	1.07	<b>1.00</b>	1.17	1.19	1.17	1.05	1.05	<b>1.04</b>	1.05	1.06	1.06

The smallest error value in each category is shown in bold for ease of comparison. The number of points in a training set is 12

<sup>a</sup>The mean values of these error metrics over 1,000 training and test sets

**Table 8** Comparison of accuracies of individual and ensemble models for the four-variable I-beam

	Individual metamodels					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
GMSE <sup>a</sup>	1.34	<b>1.00</b>	1.29	1.22	1.04	0.77	0.79	<b>0.59</b>	0.72	<b>0.59</b>	0.72
RMSE <sup>a</sup>	1.47	<b>1.00</b>	1.41	1.33	1.12	1.03	<b>1.02</b>	<b>1.02</b>	1.04	1.11	1.07

The smallest error value in each category is shown in bold for ease of comparison. The number of points in a training set is 30

<sup>a</sup>The mean values of these error metrics over 1,000 training and test sets

to be slightly less accurate than the best individual meta-model RBF.

For Fortini’s clutch problem, RBF is the most accurate individual metamodel if GMSE is used, while GP is the most accurate metamodel if RMSE is used (see Table 9). The spatial ensembles SP1 and SP3 are the most accurate ensemble models if GMSE is used, whereas the spatial ensemble SPV is the most accurate metamodel if RMSE is used. The accuracy of SPV is the same as the accuracy of the best individual metamodel.

#### 6.4 Harman function (six-variable)

The accuracies of the individual and the ensemble of metamodels constructed for six-variable Hartman function are evaluated in Table 10. GP is the most accurate individual metamodel whether evaluated by GMSE or RMSE. The spatial ensembles SP1 and SP3 are the most accurate ensemble models if evaluated by the GMSE metric, while the spatial ensembles SP1 and SP2 is the best ensemble models if evaluated by the RMSE metric. Notice that the ensembles GLO, SPV, SP1, SP2, and SP4 are more accurate than the best individual metamodel.

#### 6.5 Extended Rosenbrock function (nine-variable)

The accuracies of the individual and the ensemble of metamodels constructed for nine-variable extended Rosenbrock

function are evaluated in Table 11. RBF is the most accurate individual metamodel whether evaluated by GMSE or RMSE. The spatial ensembles SP1 and SP3 are the most accurate ensemble models if evaluated by the GMSE metric, but the global ensemble GLO is the best model if evaluated by the RMSE metric. Notice that the global ensemble GLO is more accurate than the best individual metamodel RBF.

The extended Rosenbrock function is chosen as the representative example for high-dimensional problems to analyze the effect of increasing the training points. The number of training points is increased from 110 to 220, and the analysis is repeated. The results are presented in Table 12. It is seen that as the number of training points is increased, the performances of the individual as well as the ensemble models change slightly.

#### 6.6 Dixon-Price function (12-variable)

The accuracies of the individual and the ensemble of metamodels constructed for the 12-variable Dixon-Price function are presented in Table 13. PRS is the most accurate individual metamodel whether evaluated by GMSE or RMSE. The spatial ensembles SP1 and SP3 are the most accurate ensemble models if evaluated by the GMSE metric, but the global ensemble GLO is the best model if evaluated by the RMSE metric. Notice that the global ensemble GLO is more accurate than the best individual metamodel.

To provide an overall picture of the performances of the ensemble models, Table 14 presents the three most accurate

**Table 9** Comparison of accuracies of individual and ensemble models for the Fortini’s clutch

	Individual metamodels					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
GMSE <sup>a</sup>	1.67	<b>1.00</b>	4.00	2.11	3.06	0.90	0.80	<b>0.63</b>	0.72	<b>0.63</b>	0.72
RMSE <sup>a</sup>	1.47	1.13	3.12	1.72	<b>1.00</b>	1.03	<b>1.00</b>	1.06	1.02	1.32	1.14

The smallest error value in each category is shown in bold for ease of comparison. The number of points in a training set is 30

<sup>a</sup>The mean values of these error metrics over 1,000 training and test sets

**Table 10** Comparison of accuracies of individual and ensemble models for the Hartman function

	Individual metamodells					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
GMSE <sup>a</sup>	1.25	1.12	1.00	1.04	<b>1.00</b>	0.85	0.80	<b>0.59</b>	0.70	<b>0.59</b>	0.70
RMSE <sup>a</sup>	1.26	1.11	1.03	1.07	<b>1.00</b>	0.94	0.94	<b>0.92</b>	<b>0.92</b>	1.06	0.98

The smallest error value in each category is shown in bold for ease of comparison. The number of points in a training set is 56

<sup>a</sup>The mean values of these error metrics over 400 training and test sets

**Table 11** Comparison of accuracies of individual and ensemble models for the extended Rosenbrock function

	Individual metamodells					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
GMSE <sup>a</sup>	1.11	<b>1.00</b>	1.94	1.60	2.24	0.97	0.93	<b>0.62</b>	0.77	<b>0.62</b>	0.77
RMSE <sup>a</sup>	1.09	<b>1.00</b>	1.97	1.63	2.33	<b>0.98</b>	1.15	1.12	1.16	1.41	1.31

The smallest error value in each category is shown in bold for ease of comparison. The number of points in a training set is 110

<sup>a</sup>The mean values of these error metrics over 100 training and test sets

**Table 12** Comparison of accuracies of individual and ensemble models for the extended Rosenbrock function when the number of training points is increased from 110 to 220

	Individual metamodells					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
GMSE <sup>a</sup>	<b>1.00</b>	1.02	1.95	1.71	2.65	0.97	0.94	<b>0.62</b>	0.78	<b>0.62</b>	0.78
RMSE <sup>a</sup>	1.01	<b>1.00</b>	1.86	1.65	2.61	<b>0.97</b>	1.13	1.14	1.16	1.43	1.31

The smallest error value in each category is shown in bold for ease of comparison

<sup>a</sup>The mean values of these error metrics over 20 training and test sets

**Table 13** Comparison of accuracies of individual and ensemble models for the Dixon-Price function

	Individual metamodells					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
GMSE <sup>a</sup>	<b>1.00</b>	1.25	2.33	2.43	2.33	0.96	0.97	<b>0.67</b>	0.84	<b>0.67</b>	0.84
RMSE <sup>a</sup>	<b>1.00</b>	1.25	2.41	2.47	2.41	<b>0.96</b>	1.22	1.17	1.22	1.45	1.41

The smallest error value in each category is shown in bold for ease of comparison. The number of points in a training set is 182

<sup>a</sup>The mean values of these error metrics over 20 training and test sets

**Table 14** The three most accurate ensemble models for the mathematical problems when the number of training points is twice the number of coefficients in PRS2

Note that the individual meta-models are excluded

Problem	GLO	SPV	SP1	SP2	SP3	SP4
Branin-Hoo (2 var.)	–	1	3	1	–	–
Goldstein-Price (2 var.)	2	2	1	2	–	–
I-beam design (4 var.)	3	1	1	–	–	–
Fortini’s clutch (4 var.)	2	1	–	2	–	–
Hartman (6 var.)	3	3	1	1	–	–
Extended Rosenbroock (9 var.)	1	3	2	–	–	–
Dixon-Price (12 var.)	1	3	2	3	–	–

**Table 15** Comparison of accuracies of individual and ensemble models for the intrusion distance of the automobile floor pan under offset frontal impact

	Individual metamodels					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
GMSE <sup>a</sup>	1.43	1.14	1.34	<b>1.00</b>	1.04	0.99	0.95	<b>0.74</b>	0.88	<b>0.74</b>	0.88
RMSE <sup>a</sup>	1.25	1.10	1.50	<b>1.00</b>	1.07	0.96	<b>0.94</b>	0.97	0.97	1.04	0.98

The smallest error value in each category is shown in bold for ease of comparison

<sup>a</sup>Note that only a single training set is used for the automobile crash problem

ensemble models for all the mathematical problems considered in this study. The cases when the number of training points is taken twice the number of coefficients in PRS2 are considered. Table 14 shows that the spatial ensemble models are, in general, better than the global ensemble model for low dimensional problems, while the global ensemble is the most accurate model for the high dimensional problems. The spatial ensemble based on prediction variance minimization (SPV) is usually more accurate than the proposed spatial ensembles for low-dimensional problems, while the proposed spatial ensemble SP1 becomes more accurate than SPV for high-dimensional problems.

### 6.7 Automobile crash problem

Table 15 presents a comparison of the accuracies of the individual as well as the ensemble of metamodels for the intrusion distance of the automobile floor pan under offset frontal impact. KR1 is found to be the most accurate individual metamodel whether evaluated by GMSE or RMSE. The spatial ensembles SP1 and SP3 are the most accurate ensemble models if evaluated by the GMSE metric, while the spatial ensemble SPV is the most accurate metamodel if evaluated by the RMSE metric.

The accuracies of the individual and the ensemble of metamodels built for prediction of the acceleration of the automobile floor pan under offset frontal impact are compared in Table 16. GP is the most accurate individual meta-

model if evaluated by GMSE, while RBF is best individual metamodel if evaluated by RMSE. The spatial ensembles SP1 and SP3 are the most accurate ensemble models if evaluated by the GMSE metric, while the spatial ensemble SP2 is the best metamodel if evaluated by the RMSE metric.

### 6.8 Correlation between the pointwise cross validation error and error at a prediction point

The example problems revealed that even though the spatial ensemble models show excellent performances in reducing GMSE (20–40% accuracy improvement), their performances in reducing RMSE are not that impressive (at most 4% accuracy improvement). Therefore, it is necessary to investigate whether the pointwise cross validation error is a good surrogate for the error at a prediction point. For this purpose, their correlation is computed for all the individual metamodels for the Branin-Hoo function (chosen as the representative example). Table 17 shows that the pointwise cross validation error is weakly correlated to the error at a prediction point as manifested by the small mean values and very large coefficient of variations of the correlation coefficients. It is found that the pointwise cross validation error is not a good surrogate for the error at a point, even though GMSE is a good surrogate for the global error RMSE. Similar investigations are also conducted for the other example problems and it is found that the pointwise

**Table 16** Comparison of accuracies of individual and ensemble models for the acceleration of the automobile floor pan under offset frontal impact

	Individual metamodels					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
GMSE <sup>a</sup>	1.65	1.26	1.44	1.02	<b>1.00</b>	0.99	0.99	<b>0.80</b>	0.90	<b>0.80</b>	0.90
RMSE <sup>a</sup>	1.79	<b>1.00</b>	2.00	1.11	1.18	1.14	0.99	0.98	<b>0.97</b>	1.44	1.16

The smallest error value in each category is shown in bold for ease of comparison

<sup>a</sup>Note that only a single training set is used for the automobile crash problem

**Table 17** Correlation coefficient of the square of the pointwise cross validation error and square of the error at a prediction point for the Branin-Hoo function

	PRS	RBF	KR0	KR1	GP
Mean <sup>a</sup>	0.15	0.16	0.14	0.23	0.16
COV <sup>a</sup>	1.2	1.2	1.5	0.84	1.3

<sup>a</sup>Computed over 1,000 training and test sets

cross validation error is only weakly correlated to the error at a prediction point.

### 6.9 Evaluating the accuracies of the metamodels using different error metrics

As noted earlier, the RMSE is the main error metric of interest in this paper, since the weight factor selection is based on GMSE metric. If a different error metric than GMSE was used for weight factor selection, a different error metric would be the main interest. However, it is also worthy to evaluate the accuracies of the metamodels using different error metrics. For this purpose, Branin-Hoo and the extended Rosenbrock functions are chosen as representative examples of low-dimensional and high-dimensional functions, respectively. The accuracies of the metamodels are evaluated using the following error metrics: the RMSE, the mean absolute error (MAE), the maximum absolute error (MAXE), and the coefficient of multiple determination ( $R^2$ ). The metrics RMSE, MAE, and MAXE are normalized with respect to the most accurate individual metamodel, while the  $R^2$  metric is not (since it is already a normalized value). Table 18 shows for the Branin-Hoo function that RMSE, MAE, and  $R^2$  metrics declares the same metamodels as being the most accurate one, while MAXE metric points out a different one. Table 19 shows for the extended Rosenbrock function that all the error metrics pinpoint RBF as the most accurate individual metamodel, and the GLO as the most accurate ensemble model.

**Table 18** Evaluating accuracies of individual and ensemble models for Branin-Hoo function using different error metrics

	Individual metamodels					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
RMSE	1.26	<b>1.00</b>	1.01	1.21	1.03	0.99	<b>0.96</b>	0.97	<b>0.96</b>	1.05	0.99
MAE	1.47	<b>1.00</b>	1.04	1.22	1.09	1.03	<b>0.98</b>	1.00	<b>0.98</b>	1.07	1.01
MAXE	1.02	1.05	<b>1.00</b>	1.25	1.01	<b>0.97</b>	1.00	1.00	1.00	1.05	1.02
$R^2$	0.60	<b>0.74</b>	0.72	0.59	0.70	0.74	<b>0.76</b>	0.75	<b>0.76</b>	0.71	0.75

The best performance in each category is shown in bold for ease of comparison. The number of points in a training set is 12

## 7 Conclusions

The advantages of using an ensemble of metamodels instead of a single metamodel have been investigated by several researchers. Most of these studies have used global error measures while forming the ensemble. The generalized mean square cross validation error (GMSE) has been successfully used as a global measure. In addition, the use of prediction variance, a local error measure, was also proposed to determine the weight factors of individual metamodels in the ensemble. The main objective of this paper was to investigate the efficiency of various local error measures. As an alternative to prediction variance, the use of pointwise cross validation error was also presented. The accuracies of the individual models and ensemble models were tested on various problems with varying complexity: four mathematical benchmark problems, two structural mechanics problems and an automobile crash problem (with two critical responses) requiring high-fidelity simulations of a complex model with nonlinear responses. Since the accuracies of metamodels depend on training data (size and location), test data (size and location), error metric, and the complexity of the problem, the effects of these parameters were also investigated. From the results of this study, the following conclusions could be drawn.

- In general, the spatial ensemble models showed better performances than the global ensemble for low-dimensional problems, while the global ensemble was the most accurate model for high-dimensional problems. In addition, the ensemble models were 1–6% more accurate than the best individual metamodel in seven out of nine responses appeared in all example problems.
- The accuracies of the spatial ensembles that use pointwise cross validation and the accuracy of the spatial ensemble that uses prediction variance were close. The difference between the accuracies of the spatial ensemble based on prediction variance and the most accurate spatial ensemble based on pointwise cross validation

**Table 19** Evaluating accuracies of individual and ensemble models for extended Rosenbrock function using different error metrics

	Individual metamodels					Ensemble models					
	PRS	RBF	KR0	KR1	GP	GLO	SPV	SP1	SP2	SP3	SP4
RMSE	1.09	<b>1.00</b>	1.97	1.63	2.23	<b>0.98</b>	1.15	1.12	1.16	1.41	1.31
MAE	1.09	<b>1.00</b>	2.00	1.65	2.40	<b>0.98</b>	1.14	1.13	1.17	1.34	1.27
MAXE	1.03	<b>1.00</b>	2.00	1.52	2.19	<b>0.96</b>	1.30	1.18	1.21	1.76	1.60
R <sup>2</sup>	0.77	<b>0.81</b>	0.27	0.51	0.10	<b>0.82</b>	0.75	0.77	0.75	0.63	0.68

The best performance in each category is shown in bold for ease of comparison. The number of points in a training set is 110

error were at most 4% over all problems. It is worthy to note that this observation is dependent on the accuracy criterion used.

- While computing the weight factors at a prediction point from the weight factors at data points, scaling the weight factors at all the data points proportional to the square of the distance between the prediction point and the data point was found to lead to around 2–10% more accurate predictions than assigning the weight factors at a prediction point equal to the weight factors of the closest data point.
- While computing the weight factors at data points, selecting the weight factor of the individual metamodel with smallest GMSE as one, and assigning zero weights to other metamodels is found to be better than selecting the weight factors at the data points inversely proportional to the square of the cross-validation errors to reduce GMSE errors. This practice resulted in 1–4% reduction in RMSE.
- Even though the spatial ensemble models showed excellent performances in reducing GMSE (20–40% accuracy improvement), their performances in reducing RMSE were not that impressive (at most 4% accuracy improvement). This behavior can be explained by the finding of this paper that the pointwise cross validation error is not a good surrogate for the error at a point, even though GMSE is a good surrogate for the global error.
- The effect of the training points was also investigated. For the Branin-Hoo function (chosen as representative example for low dimensional problems), it was found that as the number of training points was increased (1) the order of the accuracies of the metamodels changed, (2) the deviation between the accuracies of the metamodels increased, (3) the global ensemble became more accurate than the spatial ensembles, and (4) the best individual metamodel became more accurate than all ensemble models. For the extended Rosenbrock function (chosen as representative example for high dimensional problems), it was observed that as the number of training points was increased, the performances of the

individual as well as the ensemble models changed only slightly.

- The accuracies of the metamodels were evaluated using different error metrics for the Branin-Hoo and the extended Rosenbrock functions. It was found that RMSE, MAE, and R<sup>2</sup> metrics indicates the same metamodel as the most accurate models, while MAXE metric may lead to a different one.

**Acknowledgments** This work was partially supported by the Research Fund of TÜBİTAK (Project No. 109M537). The author thanks Prof. Raphael T. Haftka at the University of Florida for his helpful comments, and Dr. Kiran Solanki at the Mississippi State University for performing automobile crash simulations.

**Appendix: Description of selected metamodeling techniques**

A brief overview of the mathematical formulation of PRS, RBF, KR, and GP metamodeling techniques is provided here. This appendix is close to the appendix of the paper by Acar and Rais-Rohani (2009).

Polynomial response surface approximations, PRS

The most commonly used PRS model is the second-order model in the form of a second-degree algebraic polynomial function as

$$\hat{f}(x) = b_0 + \sum_{i=1}^L b_i x_i + \sum_{i=1}^L b_{ii} x_i^2 + \sum_{i=1}^{L-1} \sum_{j=i+1}^L b_{ij} x_i x_j \quad (22)$$

where  $\hat{f}$  is the response surface approximation of the actual response function,  $f$ ,  $L$  is the number of variables in the input vector  $x$ , and  $b_0, b_i, b_{ii}, b_{ij}$  are the unknown coefficients to be determined by the least squares technique.

Radial basis function, RBF

RBF methods were originally developed to approximate multivariate functions based on scattered data. For a data

set consisting of the values of input variables and response values at  $n$  sampling points, the true function  $f(\mathbf{x})$  can be approximated as

$$\tilde{f}(\mathbf{x}) = \sum_{i=1}^n \lambda_i \phi(\|\mathbf{x} - \mathbf{x}_i\|) \tag{23}$$

where  $\mathbf{x}$  is the vector of input variables,  $\mathbf{x}_i$  is the vector of input variables at the  $i$ th sampling point,  $\|\mathbf{x} - \mathbf{x}_i\| = \sqrt{(\mathbf{x} - \mathbf{x}_i)^T(\mathbf{x} - \mathbf{x}_i)}$  is the Euclidean norm representing the radial distance,  $r$  from design point  $\mathbf{x}$  to the sampling point or center  $\mathbf{x}_i$ ,  $\phi$  is a radially symmetric basis function, and  $\lambda_i$ ,  $i = 1, n$  are the unknown interpolation coefficients. Equation (23) represents a linear combination of a finite number of radially symmetric basis functions. Some of the most commonly used RBF formulations include:  $\phi(r) = r^2 \log(r)$  (thin-plate spline);  $\phi(r) = e^{-\alpha r^2}$ ,  $\alpha > 0$  (Gaussian);  $\phi(r) = \sqrt{r^2 + c^2}$  (multiquadric); and  $\phi(r) = 1/\sqrt{r^2 + c^2}$  (inverse multiquadric). The parameter  $c$  in the multiquadrics is a constant. If the  $r$  values are normalized to the range of (0,1), then  $0 < c \leq 1$ . The choice of  $c = 1$  is found to be suitable for most function approximations. The feature that makes these functions excellent candidates for  $\phi$  is not simply their radial symmetry but their smoothness and certain properties of their Fourier transform (Buhmann 2003). In this study, the multiquadric formulation of RBF is chosen because of its prediction accuracy and its commonly linear and possibly exponential rate of convergence with increased sampling points.

Given the design coordinates of  $n$  sampling points and associated responses, the unknown coefficients in (23) are found by minimizing the residual or the sum of the squares of the deviations expressed as

$$R = \sum_{j=1}^n \left[ f(\mathbf{x}_j) - \sum_{i=1}^n \lambda_i \phi(\|\mathbf{x}_j - \mathbf{x}_i\|) \right]^2 \tag{24}$$

Expressed in matrix form, (24) appears as

$$[A]\{\lambda\} = \{\mathbf{f}\} \tag{25}$$

where  $[A] = [\phi\|\mathbf{x}_j - \mathbf{x}_i\|]$ ,  $j = 1, n; i = 1, n$ ,  $\{\lambda\}^T = \{\lambda_1, \lambda_2, \dots, \lambda_n\}^T$ , and  $\{\mathbf{f}\}^T = \{f(x_1), f(x_2), \dots, f(x_n)\}^T$ . The coefficient vector  $\lambda$  is obtained by solving (25).

### Kriging, KR

The basic assumption of KR is the estimation of the response in the form

$$f(x) = p(x) + Z(x) \tag{26}$$

where  $f(x)$  is the response function of interest,  $p(x)$  is a known polynomial that globally approximates the response,

and  $Z(x)$  is the stochastic component that generates deviations such that the Kriging model interpolates the sampled response data. In this work, when  $p(x)$  is chosen as a constant, the metamodel is denoted with KR0, and when  $p(x)$  is chosen as a linear polynomial, the metamodel is denoted with KR1.

The stochastic component has a mean value of zero and covariance of

$$COV [Z(x_i), Z(x_j)] = \sigma^2 \mathbf{R} [R(x_i, x_j)] \tag{27}$$

where  $\mathbf{R}$  is  $N \times N$  correlation matrix if  $N$  is the number of data points,  $R(x^i, x^j)$  is correlation function between the two data points  $x^i$  and  $x^j$ . Mostly, the correlation function is chosen as Gaussian, that is,

$$R(\theta) = \prod_{k=1}^L \exp(-\theta_k d_k^2) \tag{28}$$

where  $L$  is the number of variables,  $d_k = x_k^i - x_k^j$  is the distance between the  $k$ th components of the two data points  $x^i$  and  $x^j$ , and  $\theta_k$  are the unknown parameters to be determined.

Once the correlation function has been selected, the response  $f$  is predicted as

$$\hat{f}(x) = \hat{\beta} + \mathbf{r}^T(x) \mathbf{R}^{-1}(\mathbf{f} - \hat{\beta} \mathbf{p}) \tag{29}$$

where  $\mathbf{r}^T(x)$  is the correlation vector of length  $N$  between a prediction point  $x$  and the  $N$  sampling points,  $f$  represents the responses at the  $N$  points and  $p$  is an  $L$ -vector of ones (in the case that  $p(x)$  is taken as a constant). The vector  $\mathbf{r}$  and scalar  $\hat{\beta}$  are given by

$$\begin{aligned} \mathbf{r}^T(x) &= [R(x, x^1), R(x, x^2), \dots, R(x, x^N)]^T, \\ \hat{\beta} &= (\mathbf{p}^T \mathbf{R}^{-1} \mathbf{p})^{-1} \mathbf{p}^T \mathbf{R}^{-1} \mathbf{f} \end{aligned} \tag{30}$$

The variance of the output model (which is different than the variance of the sampled output) can be estimated as

$$\hat{\sigma}^2 = \frac{(\mathbf{f} - \hat{\beta} \mathbf{p})^T \mathbf{R}^{-1} (\mathbf{f} - \hat{\beta} \mathbf{p})}{N} \tag{31}$$

The unknown parameters  $\theta_k$  can be estimated by solving the following constrained maximization problem (Simpson et al. 2001)

$$\begin{aligned} \text{Max } \Phi(\Theta) &= \frac{-[N \ln(\hat{\sigma}^2) + \ln |\mathbf{R}|]}{2} \\ \text{s.t. } \Theta &> 0 \end{aligned} \tag{32}$$

where  $\Theta$  is the vector of unknown parameters  $\theta_k$ , and both  $\hat{\sigma}$  and  $\mathbf{R}$  are functions of  $\Theta$ .

In this work, the MATLAB Kriging toolbox developed by Lophaven et al. (2002) is used.



Gaussian process, GP

Gaussian process assumes that the output variables  $f_N = \{f_n(x_n^1, x_n^2, \dots, x_n^L)\}_{n=1}^N$  are related to each other with a Gaussian joint probability distribution

$$P(f_N|C_N, X_N) = \frac{1}{\sqrt{(2\pi)^N |C_N|}} \exp\left[-\frac{1}{2}(f_N - \mu)^T C_N^{-1}(f_N - \mu)\right] \tag{33}$$

where  $X_N = \{x_n\}_{n=1}^N$  are  $N$  pairs of  $L$ -dimensional input variables  $x_n = (x_n^1, x_n^2, \dots, x_n^L)$ ,  $C_N$  is the covariance matrix with elements of  $C_{ij} = C(x_i, x_j)$ .  $\mu$  is the mean output vector. GP estimates the output at a prediction point  $x_p = (x_p^1, x_p^2, \dots, x_p^L)$  as

$$\hat{f}(x_p) = k^T C_N^{-1} f_N \tag{34}$$

where  $k = [C(x_1, x_p), \dots, C(x_N, x_p)]$ . One of the nice properties of the GP is that the standard deviation at the prediction point is readily available without a requirement of any extra simulations. This standard deviation can be utilized as an error measure and can be calculated from

$$\sigma_{\hat{f}(x_p)} = \kappa - k^T C_N^{-1} k \tag{35}$$

where  $\kappa = C(x_p, x_p)$ .

We notice from (34) that the GP prediction depends on the covariance matrix  $C_N$ . The elements of this matrix are calculated from

$$C_{ij} = \theta_1 \exp\left[-\frac{1}{2} \sum_{l=1}^L \frac{(x_i^{(l)} - x_j^{(l)})^2}{r_l^2}\right] + \theta_2 \tag{36}$$

$$C_{ij} = \theta_1 \exp\left[-\frac{1}{2} \sum_{l=1}^L \frac{(x_j^{(l)} - x_j^{(l)})^2}{r_l^2}\right] + \theta_2 + \delta_{ij} \theta_3 \tag{37}$$

where  $\theta_1, \theta_2, \theta_3$ , and  $r_l (l = 1, 2, \dots, L)$  are called ‘‘hyperparameters’’. Here  $\delta_{ij}$  is the Kronecker delta and  $\theta_3$  is an independent noise parameter. The hyperparameters are selected so as to maximize the logarithmic likelihood that the model prediction matches the training response data. The logarithmic likelihood function  $L$  is defined as

$$L = -\frac{1}{2} \log |C_N| - \frac{1}{2} f_N^T C_N^{-1} f_N - \frac{N}{2} \log 2\pi + \ln P(\theta) \tag{38}$$

where  $P(\theta)$  is the prior distribution of the hyperparameters. In most of the applications, there is no prior knowledge of the values of the hyperparameters, so the prior distribution is uniform. Then, the last term of (38),  $\ln P(\theta)$ , is a constant and can be taken as zero for the purpose of optimization, as done in this work.

The covariance function given in (36) defines the interpolation mode of the GP metamodel, that passes through all the training data points exactly. On the other hand, (37) defines the regression mode of the model, which allows building smoother surfaces for problems with noisy data.

With the noise of the output values filtered out, the predicted surface becomes less complex and may not pass through all the training points; however, it provides a better prediction at the non-training points. In this work, the Gaussian process code from Rasmussen and Williams (2006) is used.

References

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

Box GEP, Hunter WG, Hunter JS (1978) *Statistics for experimenters*. Wiley, New York

Buhmann MD (2003) *Radial basis functions: theory and implementations*. Cambridge University Press, New York

Dyn N, Levin D, Rippa S (1986) Numerical procedures for surface fitting of scattered data by Radial Basis Functions. *SIAM J Sci Statist Comput* 7(2):639–659

Fang H, Rais-Rohani M, Liu Z, Horstemeyer MF (2005) A comparative study of metamodeling methods for multi-objective crashworthiness optimization. *Comput Struct* 83:2121–2136

Giunta A, Watson LT (1998) A comparison of approximation modeling techniques: polynomial versus interpolating models. In: *Proceedings of the 7th AIAA/USAF/NASA/ISSMO symposium on multidisciplinary analysis & optimization*, vol 1. St. Louis, MO, pp 392–404

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

Jin R, Chen W, Simpson TW (2001) Comparative studies of metamodeling techniques under multiple modeling criteria. *Struct Multidisc Optim* 23:1–13

Lee SH, Kwak BM (2006) Response surface augmented moment method for efficient reliability analysis. *Struct Saf* 28:261–272

Lophaven SN, Nielsen HB, Søndergaard J (2002) *DACE—A MATLAB Kriging toolbox*. Informatics and mathematical modelling. Technical University of Denmark, Lyngby

MacKay DJC (1998) Introduction to Gaussian processes. In: Bishop CM (ed) *Neural networks and machine learning*. NATO ASI Series, vol 168. Springer, Berlin, pp 133–165

Martin JD, Simpson TW (2005) Use of Kriging models to approximate deterministic computer models. *AIAA J* 43(6):853–863

Messac A, Mullur AA (2008) A computationally efficient metamodeling approach for expensive multiobjective optimization. *Optim Eng* 9:37–67

Mullur AA, Messac A (2004) Extended radial basis functions: more flexible and effective metamodeling. In: *Proceedings of the 10th AIAA/ISSMO symposium on multidisciplinary analysis and optimization*. Albany, NY

- Myers RH, Montgomery DC (2002) *Response Surface Methodology: process and product optimization using designed experiments*. Wiley, New York
- Queipo NV, Haftka RT, Shyy W, Goel T, Vaidyanathan R, Tucker PK (2005) Surrogate-based analysis and optimization. *Prog Aerosp Sci* 41:1–28
- Rais-Rohani M, Solanki K, Eamon C (2006) Reliability-based optimization of lightweight automotive structures for crashworthiness. In: *Proceedings of the 11th AIAA/ISSMO multidisciplinary analysis and optimization conference*. Portsmouth, VA
- Rasmussen CE, Williams CKI (2006) *Gaussian processes for machine learning*. MIT, Cambridge
- Sacks J, Welch WJ, Mitchell TJ, Wynn HP (1989) Design and analysis of computer experiments. *Stat Sci* 4(6):409–435
- Sanchez E, Pintos S, Queipo NV (2008) Toward an optimal ensemble of kernel-based approximations with engineering applications. *Struct Multidisc Optim* 36(3):247–261
- Simpson TW, Mauery TM, Korte JJ, Mistree F (2001) Kriging models for global approximation in simulation-based multidisciplinary design optimization. *AIAA J* 39(16):2233–2241
- Stander N, Roux W, Giger M, Redhe M, Fedorova N, Haarhoff J (2004) A comparison of metamodeling techniques for crashworthiness optimization. In: *Proceedings of the 10th AIAA/ISSMO multidisciplinary analysis and optimization conference*. Albany, NY
- Viana FAC, Haftka RT, Steffen V (2009) Multiple surrogates: how cross-validation errors can help us to obtain the best predictor. *Struct Multidisc Optim* 39(6):439–457
- Wang GG, Shan S (2007) Review of metamodeling techniques in support of engineering design optimization. *ASME J Mech Des* 129(6):370–380
- Wang L, Beeson D, Wiggs G, Rayasam M (2006) A comparison of metamodeling methods using practical industry requirements. In: *Proceedings of the 47th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics, and materials conference*. Newport, RI