



An optimal pointwise weighted ensemble of surrogates based on minimization of local mean square error

Yifan Ye¹ · Zhanxue Wang¹ · Xiaobo Zhang¹

Received: 13 July 2019 / Revised: 5 January 2020 / Accepted: 17 January 2020 / Published online: 28 February 2020
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

Abstract

Surrogate models are often used as surrogates for computationally intensive simulations. And there are a variety of surrogate models which are widely used in aerospace engineering-related investigation and design. In general, there is an optimal individual surrogate for a certain research object. However, the behavior of an individual surrogate is unknown in advance. Building an ensemble of surrogates by combining different individual surrogates into a weighted-sum formulation is an efficient method to enhance the accuracy and robustness of the surrogate model. Motivated by the previous researches on the ensemble of surrogates, we propose an optimal pointwise weighted ensemble (OPWE) method, wherein the optimal pointwise weight factors are obtained based on the minimization of the local mean square error which is constructed by the global-local error (GLE). By using six well-known mathematical problems and four engineering problems, it is proved that the OPWE proposed in this paper is better than the other ensembles of surrogates in terms of both accuracy and robustness.

Keywords Ensemble of surrogates · Mean square error · Cross-validation error · Surrogate model

1 Introduction

High-fidelity computer simulations play an important role in the investigation of modern engineering systems. However, the computational cost of computer simulations is still excessive even with the current fast developed computer technology, especially for the numerical design optimization of engineering systems. It is an efficient way to reduce the computational cost by replacing the computationally expensive simulations with surrogate models in the investigation and design of complex engineering systems.

There are various surrogate models widely used in the investigation of engineering design, such as polynomial response surface (PRS) (Myers and Montgomery 2002), radial basis function (RBF) (Hardy 1971), kriging (KRG) (Sacks et al. 1989), Gaussian process (GP) (MacKay 1998), neural networks (Smith 1993), and support vector regression (SVR) (Gunn 1997).

The surrogate models are widely used in many research fields, such as surrogate-based optimization (Meng et al. 2019; Imani et al. 2018), adaptive sampling (Li et al. 2010), Bayesian control (Imani et al. 2018), and adaptive filtering (Imani et al. 2017). Generally, there is an optimal surrogate model for a specific research object. And it was proved by many researchers that one individual surrogate tends to perform quite differently for different design problems due to their underlying and dissimilar characteristics (Forrester and Keane 2009; Queipo et al. 2005; Wang and Shan 2007). Simpson et al. (2001) and Jin et al. (2001) provided recommendations on the selection of surrogates for different problems by comparing various surrogate models. However, the behavior of an individual surrogate is unknown in advance. Building an ensemble of surrogates by combining different individual surrogates into a weighted-sum formulation is an efficient method to enhance the accuracy and robustness of the surrogate model.

Responsible Editor: Shapour Azarm

✉ Xiaobo Zhang
zhangxb@nwpu.edu.cn

Yifan Ye
yyfan_@mail.nwpu.edu.cn

Zhanxue Wang
wangzx@nwpu.edu.cn

¹ Shaanxi Key Laboratory of Internal Aerodynamics in Aero-Engine, School of Power and Energy, Northwestern Polytechnical University, Xi'an, 710072, People's Republic of China

The ensemble of surrogate models can be categorized into average ensemble model and pointwise ensemble model, according to its weight calculation method:

Average ensemble model The weights of the average ensemble model stay constant over the entire design space, and the weights are calculated by using a global error measure. There are many approaches for average ensemble modeling. For example, Goel et al. (2007) proposed an average ensemble of surrogates. In their work, the generalized mean square cross-validation error (GMSE) was used to determine the weights as global data-based error measure. Acar and Rais-Rohani (2009) proposed an average ensemble of surrogates, wherein the selection of weight factors was treated as an optimization problem. Viana et al. (2009) proposed an average ensemble of surrogates based on minimization of the mean square error (MSE), and they noted that the use of weighted average surrogate does not seem to have the potential of substantial error reduction with a large number of points. Zhou et al. (2011) obtained the average weights through a recursive process. Ferreira et al. (2016) presented an approach to create ensemble of surrogates based on least squares approximation.

Pointwise ensemble model The weights of pointwise ensemble model change with the variation of prediction point, and the weights are calculated based on a local error measure. For example, Acar (2010) determined the weight factors of individual surrogate by cross-validation error, and he noted that pointwise cross-validation error is not a good surrogate for the error at a point, even though GMSE is a good representative with respect to the global error. Lee and Choi (2014) proposed to construct the ensemble of surrogates by using v nearest points. Liu et al. (2016) presented an optimal weighted pointwise ensemble. Yin et al. (2018) divided the design space into multiple subdomains, and each domain is assigned a set of optimized weight factors. Chen et al. (2018) divided the design space into two parts, and different strategies were introduced to evaluate the weight factors in each of the two parts.

The potential for average ensemble modeling to improve prediction precision is limited because of the constant weights. The optimal average ensemble model proposed by Viana et al. (2009) is one of the best average ensemble models. However, they also noted that the use of weighted average surrogates does not seem to have the potential for substantial error reduction with a large number of points. Thus, it may be a way to build an accurate pointwise ensemble model by determining the pointwise weight factors based on the minimization of the local error.

Besides, the data-based error measure, such as GMSE and cross-validation error, is widely used in the recent investigation of ensemble modeling. This is because the

data-based error measure can provide error estimation for any individual surrogate. The local data-based error measure plays an essential role in building the pointwise ensemble model. Ideally, if an accurate local data-based error measure to identify the error between the individual surrogate model and the real model is available, the pointwise ensemble model must be more accurate than the average ensemble model because the ensemble model can adapt to the local characteristics of each individual surrogate. In previous research, however, the pointwise ensemble model did not exhibit obvious advantage over the average ensemble model as there is no sufficiently accurate local data-based error measure. Thus, one could be able to enhance the accuracy of the pointwise ensemble model by improving the accuracy of the data-based local error measure.

Motivated by these findings, we propose an optimal pointwise weight ensemble (OPWE) based on the minimization of the local mean square error (LMSE). The LMSE is proposed in this paper as criteria for the error of the region near the observed points. Meanwhile, a new local data-based error measure, namely the global-local error (GLE), is proposed in this paper to construct the LMSE.

The remainder of this paper is organized as follows. In Section 2, we review the existing ensembles of surrogates. In Section 3, we describe the proposed ensemble of surrogates in detail. In Section 4, we show the example problems. In Section 5, we present the numerical procedure. The results and discussion appear in Section 6, followed by the summary of important conclusions in Section 7.

2 Existing ensembles of surrogates

2.1 The average ensemble of surrogates proposed by Goel et al. (2007)

Goel et al. used the GMSE as the global error measure to determine the weights using a heuristic formulation:

$$\begin{aligned} w_i &= \frac{w_i^*}{\sum_{j=1}^n w_j^*} \\ w_i^* &= (E_i + \alpha E_{avg})^\beta \\ E_{avg} &= \frac{\sum_{i=1}^n E_i}{n} \\ \beta &< 0, \alpha < 1 \end{aligned} \quad (1)$$

where w_i denotes the weight of the i th surrogate model, n is the total number of the individual surrogates, and E_i denotes the GMSE of the i th surrogate model, which is obtained from the leave-one-out strategy and is calculated as

$$E_i = \frac{\sum_{k=1}^m (y(k) - \hat{y}(k))^2}{m} \quad (2)$$

In the equation, m is the number of points in the training set, $y(k)$ is the true response at the k th observed point, and $\hat{y}(k)$ is the corresponding predicted value from the individual surrogate constructed by using all except the k th observed points.

There are two user-defined parameters α and β in this method, which are employed to control the importance of averaging surrogate and the importance of individual surrogate, respectively. Goel et al. (2007) set $\alpha = 0.05$ and $\beta = -1$ in their study.

2.2 The average ensemble of surrogates proposed by Acar and Rais-Rohani (2009)

Acar and Rais selected weights by solving an optimization problem of the form

$$\begin{aligned}
 & \text{Find } w_i \\
 & \text{Minimize } GMSE \\
 & \text{s.t. } \sum_{i=1}^n w_i = 1
 \end{aligned} \tag{3}$$

where $GMSE$ is the generalized mean square cross-validation error of the ensemble model, w_i is the weight of the i th individual surrogate, and n is the total number of the individual surrogates.

2.3 The average ensemble of surrogates proposed by Vinan et al. (2009)

Viana et al. selected weights by following an approach based on the minimization of the mean square error (MSE):

$$MSE = \frac{\int_V e^2(\mathbf{x})}{V} = \mathbf{w}^T \mathbf{C} \mathbf{w} \tag{4}$$

where the function $e(\mathbf{x})$ is the error of the ensemble model at \mathbf{x} , \mathbf{w} is the vector of weights of the individual surrogates, and V denotes the sample space. They approximated the value of \mathbf{C} by using the vectors of cross-validation errors, e^* , as

$$c_{ij} \simeq \frac{\mathbf{e}_i^{*T} \mathbf{e}_j^*}{m} \tag{5}$$

where m is the number of observed points.

The weights are obtained by minimizing the MSE, as

$$\begin{aligned}
 & \text{Find } \mathbf{w} \\
 & \text{Minimize } MSE_{WAS} = \mathbf{w}^T \mathbf{C} \mathbf{w}
 \end{aligned} \tag{6}$$

which is subject to

$$\mathbf{1}^T \mathbf{w} = 1 \tag{7}$$

The solution is obtained by using Lagrange multipliers, as

$$\mathbf{w} = \frac{\mathbf{C}^{-1} \mathbf{1}}{\mathbf{1}^T \mathbf{C}^{-1} \mathbf{1}} \tag{8}$$

The solution of (8) may include negative weights as well as weights with values larger than one. Viana et al. suggested to solve (8) using only the diagonal elements of \mathbf{C} to enforce the positivity.

2.4 The pointwise ensemble of surrogates proposed by Acar (2010)

Acar proposed that the weight of the individual surrogate with the smallest cross-validation error is one while the other individual surrogates have zero weights. And the spatial weights are calculated by

$$\begin{aligned}
 w_i(\mathbf{x}) &= \sum_{k=1}^n W_{i,k} I_k(\mathbf{x}) \\
 I_k(\mathbf{x}) &= \frac{1}{d_k^2(\mathbf{x})} \\
 d_k(\mathbf{x}) &= \|\mathbf{x} - \mathbf{x}_k\|
 \end{aligned} \tag{9}$$

where $W_{i,k}$ is the pointwise weight of the i th surrogate at the k th observed point. $W_{i,k}$ is equal to one for the individual surrogate with the smallest cross-validation error at an observed point, and it is equal to zero for all other individual surrogates. $d_k(\mathbf{x})$ is the Euclidian distance between \mathbf{x} and \mathbf{x}_k , and n is the total number of the individual surrogates.

3 The proposed ensemble of surrogates

A pointwise ensemble of surrogates named optimal pointwise weighted ensemble (OPWE) is proposed in this section. The pointwise weight of each individual surrogate at an observed point is determined by minimizing the local error of the region around this observed point. The local mean square error (LMSE) is defined in this section to measure the local error of the region around the observed point. And a new local data-based error measure named global-local error (GLE) is proposed in this section to construct the LMSE.

3.1 Determining the weights at observed points by minimizing the local error

Viana et al. (2009) determined the global weight factors by minimizing the MSE. In their work, the MSE represents the global error of the ensemble model. And we determine the pointwise weight factors at an observed point by minimizing the local error of the region around this observed point. And the LMSE is used to measure this local error:

$$LMSE_{WAS}(\mathbf{x}) = \frac{\int_{V'} e^2(\mathbf{x}) d\mathbf{x}}{V'} = \mathbf{w}(\mathbf{x})^T \mathbf{C}(\mathbf{x}) \mathbf{w}(\mathbf{x}) \tag{10}$$

The formulation of LMSE is similar to MSE, except that the weight and \mathbf{C} change with \mathbf{x} , where V' denotes the region around the observed point and \mathbf{C} is the error matrix. And the weights are calculated as

$$\mathbf{w}(\mathbf{x}) = \frac{(\mathbf{C}(\mathbf{x})^{-1}\mathbf{1})}{(\mathbf{1}^T\mathbf{C}(\mathbf{x})^{-1}\mathbf{1})} \quad (11)$$

Similar to (8), the solution of (11) may include negative weights as well as weights with values larger than one. Viana et al. suggested to solve (8) using only the diagonal elements of \mathbf{C} to enforce positivity. In this paper, we solve (11) by using the diagonal elements of \mathbf{C} if the solution based on the full elements of \mathbf{C} include negative weights or weight values larger than one.

3.2 Approximating the local error of the region around the observed points

The local error of the region around an observed point is not only related to the error of the observed point, but also related to the error of the nearby observed points. Thus, we approximate the elements of the \mathbf{C} matrix by using an error measure vector. The elements of \mathbf{C} matrix are defined in (12), as

$$c_{ij}(\mathbf{x}) = \frac{\mathbf{e}_i^*(\mathbf{x})^T \bullet \mathbf{e}_j^*(\mathbf{x})}{v} \quad (12)$$

where $\mathbf{e}_i^*(\mathbf{x})$ is the local error measure vector of the i th surrogate at \mathbf{x} , and it represents the local error of the region around \mathbf{x} . The parameter v denotes the vector dimension of $\mathbf{e}_i^*(\mathbf{x})$. The local error measure vector of an observed point is constructed with the local error measure of nearby observed points. Meanwhile, the distances between observed points are used to correct the local error measure vector. A new local error measure named GLE, which combines the global error measure and the local error measure, is employed to prevent the bias in error estimation of the local error measure.

The details regarding the calculation of $\mathbf{e}_i^*(\mathbf{x})$ at each observed point will be shown later in this section.

3.2.1 The combination of the global error measure and the local error measure

Some researchers built average ensemble models using GMSE, and GMSE is a good criterion for the global error to determine weights in the average ensemble model. It is difficult to further improve the accuracy of the average ensemble model due to the constant weights. There are also some researches who built pointwise ensemble models using pointwise cross-validation error. The accuracy of the pointwise ensemble model may improve due to the pointwise weights. However, the improvement depends on

the accuracy of the local error measure. The accuracy of the pointwise ensemble model may be poor only if the local error measure at an observed point is inaccurate although the local error measure at the other observed points is accurate.

In this paper, we combine the GMSE and the pointwise cross-validation error to enhance the accuracy of the local error measure, as

$$e_i(\mathbf{x}) = GMSE_i \bullet CV_i(\mathbf{x}) \quad (13)$$

where $e_i(\mathbf{x})$ is the local error measure of the i th surrogate at \mathbf{x} , $GMSE_i$ is the generalized mean square cross-validation error of the i th surrogate, and $CV_i(\mathbf{x})$ is the pointwise cross-validation error of the i th surrogate at \mathbf{x} .

A similar form of the formula in (13) was first put forward by Liu et al. (2016). Liu et al. proposed that the weight at an observed point of the individual surrogate with the smallest absolute value of $e_i(\mathbf{x})$ is equal to one while the other individual surrogates have zero weights. The absolute value of $e_i(\mathbf{x})$ is used to identify the individual surrogate with the best accuracy at an observed point. In other words, only the order of the $e_i(\mathbf{x})$ is used to determine the weights. In the current work, however, the value of $e_i(\mathbf{x})$ is used to calculate weights. In the approach of Liu et al., one only needs to be able to precisely identify the most accurate individual surrogate based on $e_i(\mathbf{x})$. In the current approach, however, the value of $e_i(\mathbf{x})$ needs to be able to reflect the local error of the i th surrogate at \mathbf{x} .

3.2.2 Use of the nearby observed points

The local error measure vector of an observed point is constructed with the local error measure of nearby observed points. On the one hand, the local error measure vector at an observed point represents the local error of the region around this observed point. Thus, all the observed points in the region should be utilized to construct the local error measure. On the other hand, constructing the local error measure vector using nearby points is also a way to prevent the bias in error estimation of the local error measure.

Thus, $\mathbf{e}_i^*(\mathbf{x})$ at each observed point is calculated by

$$\mathbf{e}_i^*(\mathbf{x}) = \begin{bmatrix} e_i(\mathbf{x}_1) \bullet l_i(\mathbf{x}_1) \\ e_i(\mathbf{x}_2) \bullet l_i(\mathbf{x}_2) \\ \dots \\ e_i(\mathbf{x}_j) \bullet l_i(\mathbf{x}_j) \\ \dots \\ e_i(\mathbf{x}_m) \bullet l_i(\mathbf{x}_m) \end{bmatrix} \quad (14)$$

where $\mathbf{e}_i^*(\mathbf{x})$ is the error measure vector of the i th surrogate at \mathbf{x} , $e_i(\mathbf{x}_j)$ is the local error measure of the i th surrogate

at x_j , and m is the total number of observed points. where $l_i(\mathbf{x}_j)$ is the weight of the i th surrogate at \mathbf{x}_j :

$$l_i(\mathbf{x}_j) = \begin{cases} 1 & d(\mathbf{x}_j, \mathbf{x}) = d_{low} \\ \bar{d}_{inv} & d_{low} < d(\mathbf{x}_j, \mathbf{x}) \leq d_{high} \\ 0 & d_{high} < d(\mathbf{x}_j, \mathbf{x}) \end{cases}$$

$$d_{low} = \min_{1 \leq j, k \leq n; j \neq k} d(\mathbf{x}_j, \mathbf{x}_k)$$

$$d_{high} = \text{average } d(\mathbf{x}_j, \mathbf{x}_k)_{1 \leq j, k \leq n; j \neq k} \tag{15}$$

where $d(\mathbf{x}_j, \mathbf{x})$ is the Euclidean distance between the observed points \mathbf{x}_j and \mathbf{x} , \bar{d}_{inv} denotes the inverse-distance relative conversion value, and d_{low} and d_{high} denote the upper and lower bounds of distance, respectively. In particular, d_{low} is defined as the minimum value of the distance among all the observed points, and d_{high} is defined as the average value of the distance among all the observed points.

For each observed point \mathbf{x} , the value of $l_i(\mathbf{x}_j)$ at the observed point \mathbf{x}_j is equal to one if the distance between \mathbf{x}_j and \mathbf{x} is less than d_{low} , the value of $l_i(\mathbf{x}_j)$ is zero if the distance between \mathbf{x}_j and \mathbf{x} is greater than d_{high} , and the value of $l_i(\mathbf{x}_j)$ is set as \bar{d}_{inv} if the distance between \mathbf{x}_j and \mathbf{x} is within the upper and lower bounds of distance.

The \bar{d}_{inv} satisfies

$$\bar{d}_{inv} = \begin{cases} 1 & d(\mathbf{x}_j, \mathbf{x}) = d_{low} \\ 0 & d(\mathbf{x}_j, \mathbf{x}) = d_{high} \end{cases}$$

$$\bar{d}'_{inv} = \begin{cases} 0 & d(\mathbf{x}_j, \mathbf{x}) = d_{low} \\ 0 & d(\mathbf{x}_j, \mathbf{x}) = d_{high} \end{cases} \tag{16}$$

where \bar{d}'_{inv} is the first derivative function of \bar{d}_{inv} .

Thus, the following third-order polynomial is employed to calculate \bar{d}_{inv} in this paper:

$$\bar{d}_{inv} = 2\bar{d}(\mathbf{x}_j, \mathbf{x}_k)^3 - 3\bar{d}(\mathbf{x}_j, \mathbf{x}_k)^2 + 1$$

$$\bar{d}(\mathbf{x}_j, \mathbf{x}_k) = \frac{(d(\mathbf{x}_j, \mathbf{x}_k) - d_{low})}{d_{high} - d_{low}} \tag{17}$$

3.3 The optimal pointwise weighted ensemble

Figure 1 shows the flowchart of the proposed OPWE. And the following points need to be explained:

1. The method described in Section 3.2 is employed in this paper to calculate the error measures (LMSE) at observed points. This method combines the global error measure and the local error measure based on the formula in (13). The nearby points are used to construct the local error measure vector to prevent the negative influence from inaccurate pointwise cross-validation error at an observed point.

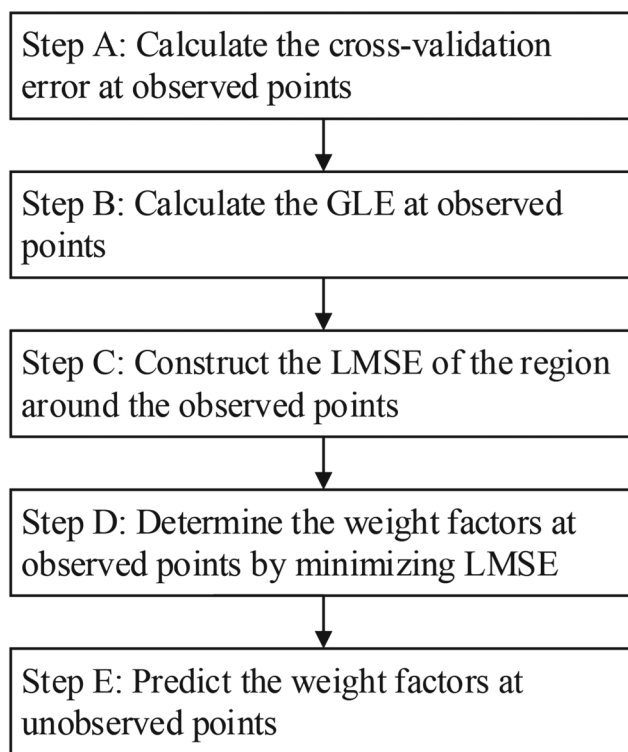


Fig. 1 The flowchart of the proposed OPWE

2. The method described in Section 3.1 is adopted in this paper to determine the weights at observed points. The LMSE is proposed to represent the local error of the region around the observed point, and the OPWE is obtained from the minimization of the LMSE as described in (10). In addition, the solution of (10) is obtained by using Lagrange multipliers as (11).
3. The weights at unobserved points are predicted by referring to the inverse-distance weighted formulas (9).

3.4 The pointwise ensemble models based on GLE

In order to verify the performance of OPWE, two pointwise ensemble models based on GLE are proposed in this paper and they are compared to the other ensemble models in the numerical procedure.

Ensemble model A Ensemble model A is a pointwise ensemble model. In this model, the weight of the individual surrogate with the smallest local error measure is set equal to one while the other individual surrogates have zero weights. The local error measure used in this model is the GLE proposed in Section 3.2.1. The weights at unobserved points are predicted according to the inverse-distance weighted formulas (9).

Ensemble model B Ensemble model B is also a pointwise ensemble model. The weights at observed points are calculated by

$$w_i(\mathbf{x}) = \frac{e_i^2(\mathbf{x})}{\sum_{j=1}^n e_j^2(\mathbf{x})} \tag{18}$$

where n is the total number of the individual surrogates and $e_i(\mathbf{x})$ is the local error measure of the i th surrogate at \mathbf{x} , and the local error measure is the GLE proposed in Section 3.2.1. The weights at unobserved points are predicted according to the inverse-distance weighted formulas (9).

Summary The ensemble models A and B are similar to the ensemble models proposed by Acar (2010) except for the local error measure. Ensemble model A is also similar to the ensemble model proposed by Liu (2016) except for the inverse-distance weighted formulation.

In ensemble model A, the order of GLE is used to determine the best individual surrogate model at observed points, and the weight of the best individual surrogate is set equal to one while the other individual surrogates have zero weights. And in ensemble model B, the value of GLE is used to calculate the weight factors at observed points. Ideally, ensemble model A is more accurate than ensemble model B if the GLE is accurate enough to determine the best individual surrogate. However, one may obtain bad weights if there are some cases with bias in error estimation of GLE. In such cases, ensemble model B is more robust than ensemble model A since the weights in model B are calculated by using the local error measure of all individual surrogates. Thus, one obtains better weights with ensemble model B than ensemble model A if there is obvious bias in error estimation of the local error measure.

4 Reference example problems

To verify the performance of the proposed ensemble model in this paper, we consider six well-known mathematical problems and four engineering problems.

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

i	a_{ij}						c_i	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	0.665
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

4.1 Mathematical problems

4.1.1 Branin-Hoo function (two variables)

$$y(x_1, x_2) = \left(x_2 - \frac{5.1x_1^2}{4\pi^2} + \frac{5x_1}{\pi} - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10 \tag{19}$$

where $x_1 \in [-5, 10]$ and $x_2 \in [0, 15]$.

4.1.2 Camelback function (two variables)

$$y(x_1, x_2) = \left(4 - 2.1x_1^x + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + (-4 + 4x_x^x)x_x^x \tag{20}$$

where $x_1 \in [-3, 3]$ and $x_2 \in [-2, 2]$.

4.1.3 Glodstein-Price function (two variables)

$$y(x_1, x_2) = [1 + (x_1 + x_2 + 1)^2 \times (19 - 4x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)] \tag{21}$$

where $x_1, x_2 \in [-2, 2]$.

4.1.4 Hartman function (six variables)

$$y(\mathbf{x}) = -\sum_{i=1}^m c_i \exp\left[-\sum_{j=1}^n a_{ij} (x_j - p_{ij})^2\right] \tag{22}$$

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

4.1.5 Extended Rosenbrock function (nine variables)

$$y(\mathbf{x}) = -\sum_{i=1}^{m-1} \left[(1 - x_i)^2 + 100(x_{i+1} - x_i^2)^2 \right] \tag{23}$$

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

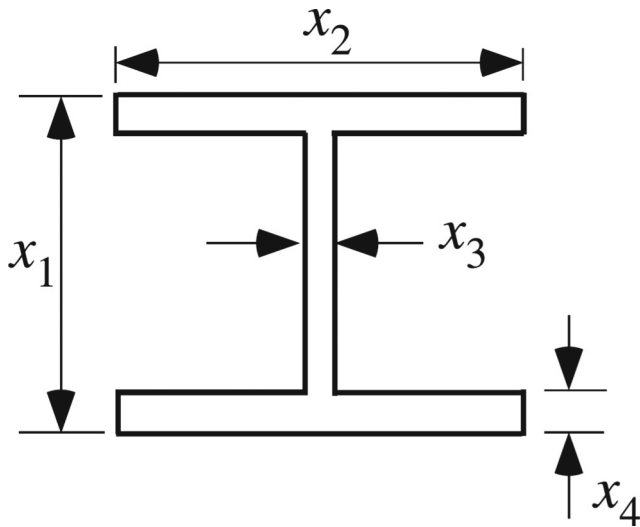


Fig. 2 The cross-section of the four variable I-beam design

4.1.6 Dixon-Price function (twelve variables)

$$y(\mathbf{x}) = (x_1 - 1)^2 + \sum_{i=2}^m i [2x_i^2 - x_{i-1}]^2 \tag{24}$$

where $x_i \in [-10, 10]$. Here, the 12-variable model ($m=12$) of Dixon-Price function is considered.

4.2 Engineering problems

4.2.1 Four variable I-beam

This four variable I-beam problem (see Fig. 2) is taken from Messac and Mullur (2008). The response for this problem is

the maximum bending stress developed in the beam, which is calculated from

$$\sigma_{max} = \frac{P x_1}{2 I} \tag{25}$$

$$I = \frac{1}{12} [x_2 x_1^3 - (x_2 - x_3) (x_1 - 2x_4)^3]$$

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

4.2.2 Fortini's clutch

The Fortini's clutch problem (see Fig. 3) is taken from Lee and Kwak (2006). The response for this problem is the contact angle, which is calculated from

$$y = \arccos \left[\frac{x_1 + 0.5(x_2 + x_3)}{x_4 - 0.5(x_2 + x_3)} \right] \tag{26}$$

The ranges of the design variables are shown in Table 2.

4.2.3 Level nozzle

A finite volume (FV) model of a Level nozzle shown in Fig. 4 is used for flow simulation using the commercial software of CFD, FLUENT. In this example, surrogate models are constructed to estimate the thrust and mass flow rate. The responses are calculated from

$$q_m = \int_{A_o} (\rho v) dA$$

$$F = \int_{A_o} [\rho v^2 + (p - p_0)] dA \tag{27}$$

where, F denotes the thrust, q_m denotes the mass flow rate, A_o denotes the nozzle outlet, ρ is the flow density at nozzle outlet, v is the flow velocity at nozzle outlet, p is the flow static pressure at nozzle outlet, and p_0 is the standard atmospheric pressure. The input variables include the nozzle

Fig. 3 The clutch assembly

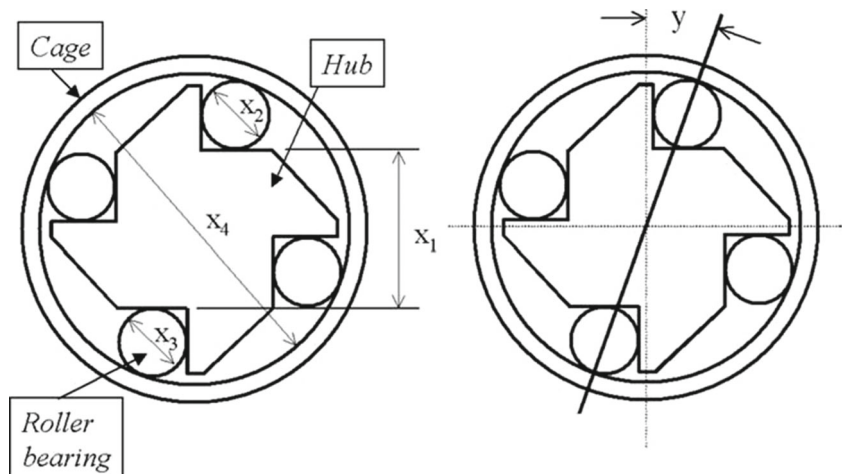


Table 2 The ranges of the design variables for Fortini's clutch problem

Variable name	Low limit	Up limit
x_1	55.0917	55.4883
x_2	22.8492	22.8708
x_3	22.8492	22.8708
x_4	101.4017	101.7983

pressure ratio (NPR, the ratio of total pressure at nozzle inlet to standard atmospheric pressure) and expansion angle (α , see in Fig. 4). And the ranges of input variables are taken as $2 \leq NPR \leq 15$ and $20 \leq \alpha \leq 50$.

4.2.4 Variable cycle engine

The variable cycle engine problem (VCE, see Fig. 5) is taken from Zhang et al. (2016). The thrust and specific fuel consumption (sfc) under subsonic cruise condition (Flight Mach number=0.9, Flight altitude=11000m) are the responses for this problem. The ranges of design variables are shown in Table 3.

5 Numerical procedure

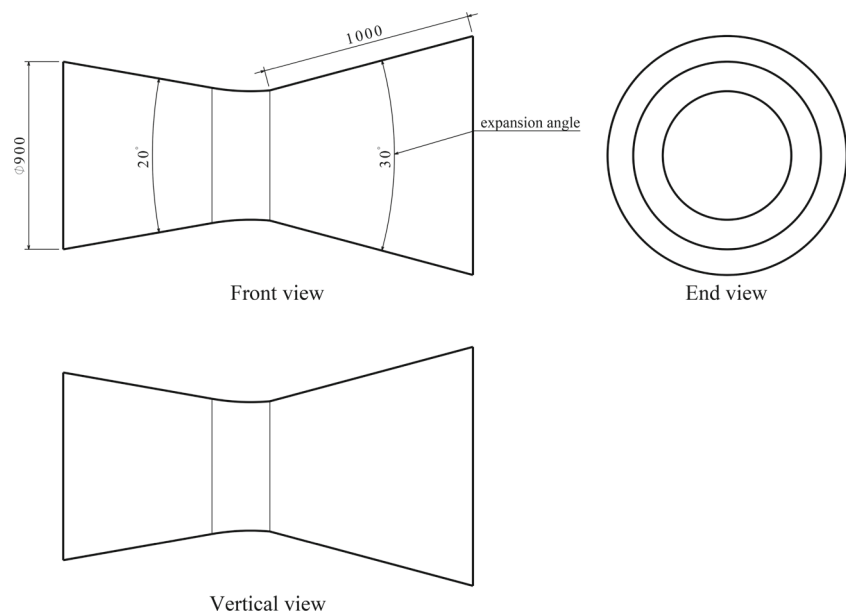
The numerical procedure is similar to the publicly available document (Acar 2010) to better evaluate and study the improvement of OPWE. The example problems are considered in this paper with varying dimensions (from 2 to 12) and selected numbers of training points. For all example problems, the Latin hypercube sampling (LHS) technique is adopted to generate the training points. The training points

are generated with the MATLAB routine "lhsdesign" and "maximin" criterion with the maximum iteration number of 1000. And the computational time of building each ensemble model is calculated with the MATLAB routine "tic" and "toc." Meanwhile, random sampling is utilized to generate 1000 test points for each training set. It is noted that the performance of the surrogate model may vary as the training set changes. Thus, some training and test sets are generated to reduce the effect of random sampling. The number of the training and test sets decreases as the dimension of the test problem increases to save the computational cost. In Table 4, we summarized the training and test data used in each problem.

Four different individual surrogates are considered: PRS, RBF, KRG, and GP. We used the SURROGATES Toolbox (2011) to build these individual surrogates. The SURROGATES Toolbox fits the kriging model using the DACE toolbox of Lophaven et al. (2002), fits the Gaussian process model using the GPML toolbox of Rasmussen and Williams (2006), and fits the radial basis function model using the RBF toolbox of Jekabsons (2009). In the case of kriging, both the constant trend model, KRG0, and the linear trend model KRG1 are used. Therefore, the ensemble of surrogates is composed of five members. The PRS surrogate is represented by fully quadratic polynomial. The RBF surrogate is based on multi-quadric formulation with the constant $c = 1$. In the kriging surrogates KRG0 and KRG1, a Gaussian correlation function is used. The covariance function in the GP surrogate is selected as the squared exponential function with automatic relevance determination distance measure.

In order to explore the effect of the number of individual surrogates, three tests with different individual surrogates

Fig. 4 Three views of Level nozzle



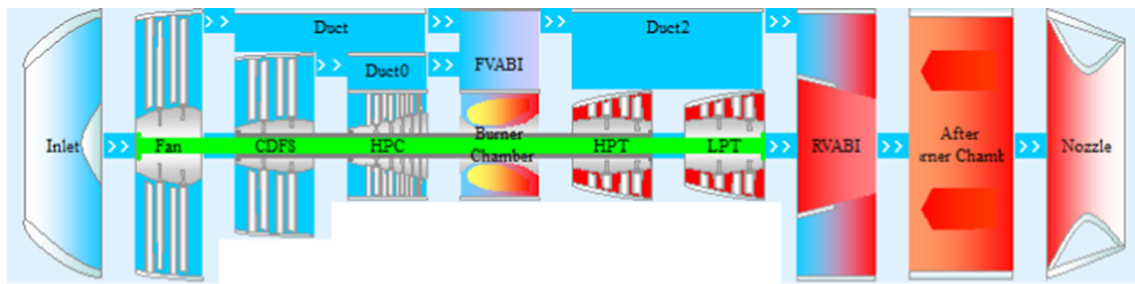


Fig. 5 Variable cycle engine

are considered. The number of individual surrogates in the three tests is 5, 4, and 3, respectively. In Table 5, we show the individual surrogates used in each test.

All the ensemble models considered in the numerical procedure have already been described in detail. The ensemble models based on the heuristic method by Goel et al. (2007) is labeled as EG; the spatial model by Acar (2010) is labeled as SP; the OWSdiag by Viana (2009) is labeled as Od; the OPWE proposed in the current work is labeled as OP; and the two other ensemble models described in Section 3.4 are labeled as EA and EB, respectively.

6 Results and discussion

In this section, the root mean square error (RSME) is chosen as the error metric of interest. The error values are normalized with respect to the most accurate individual surrogate to provide a better comparison of different models. And the most accurate individual surrogates and ensemble models are marked in bold.

In Table 6, we show the mean values of RMSE over various training and test sets for all mathematical test problems. Wherein, the numbers in the bracket are the rank of the accuracy of ensemble models. And in

Table 3 The ranges of the design variables for VCE problem

Variable name	Low limit	Up limit
Fan design bypass ratio	0.1	0.7
Fan design pressure ratio	3	4.5
CDFS design bypass ratio	0.2	0.5
CDFS design pressure ratio	1.2	1.8
HPC design pressure ratio	5	9
Burner outlet temperature(K)	1650	1950
Fan relative speed	0.7	1
CDFS inlet guide vane angle	0	45
LPT nozzle area (%)	70	130
FVABI inner duct area (%)	-50	50
RVABI inner duct area (%)	-50	50

Table 10, we show the computational time among different ensemble models for all mathematical test problems. Wherein, the numbers in the bracket are the normalized computational time, which is normalized with respect to the ensemble model with the fastest modeling speed. Tables 7 and 8 show the comparison of normalized RMSE of individual surrogate and ensemble of surrogates for all mathematical test problems with varying numbers of individual surrogates. Table 9 shows the comparison of normalized RMSE of individual surrogate and ensemble of surrogates for all engineering problems. Wherein, the numbers in the bracket are the rank of the accuracy of ensemble models.

By comparing the accuracy of individual surrogates with the different number of points in the training set, it is revealed that the best individual surrogate may change as the number of points in the training set increases for the same test problem. This indicates that the accuracy of individual surrogate depends heavily on the training set. The sequential sampling optimization is an efficient optimization framework using surrogate. The sequential sampling procedure continuously generates new observed points until the end of optimization, and of course, it causes a change in the training set. Thus, it is not suitable to use only a certain surrogate model in sequential sampling optimization. However, it is noted that using the ensemble model is much more time-consuming.

As the number of points in the training set increases, the deviation of accuracy for the most accurate individual surrogate and the ensemble models increases in most test problems. However, SP, EA, EB, and OP are more robust than Od. This is because the accuracy of local error measure continuously increases as the number of points in the training set increases. It is also noted that OP exhibits excellent stability along with the increase of the number of points in the training set in comparison with the other ensemble models.

EA is more accurate than SP in most test problems. The difference between EA and SP is only the local error measure. This indicates that GLE is a more accurate criterion for the local error than the cross-validation error. It

Table 4 Summary of training and test data used in each test problem

Problem	Number of dimension	Number of points in a training set	Number of points in a test set	Number of training and test sets
Branin-Hoo (BH)	2	12/24/36	1000	1000
Camelback (Ca)	2	12/24/36	1000	1000
Glodstein-Price (GP)	2	12/24/36	1000	1000
Hartman (Ha)	6	56/112	1000	300/200
ExtendedRosen (ER)	9	110/220	1000	100/50
DixonPrice (DP)	12	182/364	1000	30/20
I-beam	4	30	1000	1000
Fortini's clutch	4	30	1000	1000
Lavel nozzle	2	12	30	1
VCE	11	100	50	1

is also noted that for several low-dimensional test problems (dimension ≤ 5), SP shows better accuracy than EA. The cross-validation error provides the local information, and the GLE provides the local and global information. If the cross-validation error is an accurate enough criterion for the local error for a certain test problem, the global information provided by GLE is useless or even harmful for constructing an accurate ensemble model. For example, for the Nozzle flow rate (12.) test problem, SP is more accurate than EA. And it is found that SP is also more accurate than the best individual surrogate obviously. This indicates that the cross-validation error is accurate enough to represent the local error for this test problem.

EB is more accurate than EA in most low-dimensional test problems. This indicates that there is bias in error estimation of GLE. This bias raises the probability for EA to gain bad ensemble models. The deviation in accuracy between EB and EA is enlarged as the number of individual surrogate increases. This is because the increase of the number of individual surrogate raises the probability for bias in error estimation of GLE. It is also noted that EA is more accurate than EB in most of high-dimensional test problems (dimension ≤ 6). This indicates that GLE is an accurate criterion of local error for these test problems.

OP is more accurate than EA and EB in most test problems. This implies that using OP with GLE is better than EA and EB in determining the weight factors. The pointwise weights are calculated by minimizing LMSE in OP, wherein LMSE is defined as a representative for the

error of the region near the observed points. This method not only offers more efficient use of local error measure but also prevents the bias in error estimation of GLE. Altogether, OP can be utilized to prevent the bias in error estimation of GLE for local error and to improve the accuracy and robustness of ensemble models due to the efficient use of GLE. It is also noted that for several low-dimensional test problems, EB shows better accuracy than OP. In EB and OP, the GLE provides the local information of an observed point, and the LMSE provides the local information of several observed points. Thus, if the local information of an observed point can reflect the local error accurately at observed points, the local information of the other observed points provided by LMSE is useless or even harmful for constructing an accurate ensemble model.

It can be seen from Table 10 that the computational time of building OP is more than that of building the other ensemble models considered in this paper.

Except for building the individual surrogate model, the most time-consuming procedure in building the ensemble models considered in this paper (EG, Od, SP, EA, EB, and OP) is calculating the surrogate error measure. The data-based error measure is used in these ensemble models. The individual surrogate need to rebuilt n times to provide the cross-validation error for data-based error measure, where the n is the number of points in the training set. Thus, the computational time of calculating the surrogate error measure depends on the type of individual surrogate, the dimension of test problem, and the number of points in the training set.

Another time-consuming procedure in OP is calculating the distance between each observed point. The computational time of this procedure depends on the dimension of the test problem and the number of points in the training set.

As the number of points in the training set increases, the computational time of computing surrogate error measure increases. Thus, the computational time of building

Table 5 The individual surrogates used in each test

	PRS	RBF	KRG0	KRG1	GP
Test 1	✓	✓	✓	✓	✓
Test 2	✓	✓	✓	✓	×
Test 3	✓	✓	✓	×	×

Table 6 Comparison of the normalized RMSE of individual surrogate and ensemble of surrogates for the all mathematical test problems with five individual surrogates

RMSE ^a	Individual surrogate					Ensemble of surrogates					
	RBF	KRG0	PRS	KRG1	GP	EG	Od	SP	EA	EB	OP
BH(12.)	1.24	<i>1.00</i>	1.56	1.01	3.11	1.13(5)	1.07(2)	1.13(5)	1.09(4)	1.07(2)	<i>1.06(1)</i>
BH(24.)	<i>1.00</i>	1.42	1.78	1.47	3.28	1.07(6)	1.03(5)	1.00(2)	1.01(4)	<i>0.98(1)</i>	1.00(2)
BH(36.)	<i>1.00</i>	1.57	2.32	1.58	3.88	1.09(6)	1.01(5)	0.97(3)	0.97(3)	<i>0.95(1)</i>	0.96(2)
Ca(12.)	1.02	1.25	<i>1.00</i>	1.29	1.43	1.11(3)	1.10(2)	1.13(6)	1.12(5)	1.11(3)	<i>1.09(1)</i>
Ca(24.)	<i>1.00</i>	1.58	1.27	1.63	1.61	1.17(3)	<i>1.15(1)</i>	1.21(6)	1.18(5)	1.17(3)	<i>1.15(1)</i>
Ca(36.)	1.44	<i>1.00</i>	2.47	1.03	2.52	1.21(5)	1.11(3)	1.21(5)	1.12(4)	1.10(2)	<i>1.05(1)</i>
GP(12.)	1.01	<i>1.00</i>	1.25	1.02	1.17	<i>1.00(1)</i>	<i>1.00(1)</i>	1.03(5)	1.04(6)	1.01(3)	1.01(3)
GP(24.)	1.27	<i>1.00</i>	2.00	1.02	1.44	1.10(6)	1.09(5)	1.08(3)	1.09(4)	1.07(2)	<i>1.06(1)</i>
GP(36.)	1.70	<i>1.00</i>	3.73	1.03	1.79	1.23(6)	1.18(5)	1.12(4)	1.11(3)	<i>1.08(1)</i>	<i>1.08(1)</i>
Ha(56.)	1.21	<i>1.00</i>	1.37	1.04	1.46	1.05(6)	1.03(4)	1.03(4)	<i>1.01(1)</i>	1.02(3)	<i>1.01(1)</i>
Ha(112.)	1.25	<i>1.00</i>	1.36	1.03	1.72	1.04(5)	1.03(4)	1.04(5)	1.02(2)	1.02(2)	<i>1.01(1)</i>
ER(110.)	<i>1.00</i>	2.03	1.08	1.57	4.44	1.19(6)	1.05(4)	1.09(5)	1.01(2)	1.03(3)	<i>0.98(1)</i>
ER(220.)	1.02	2.12	<i>1.00</i>	1.61	5.16	1.22(6)	1.06(4)	1.13(5)	1.03(2)	1.05(3)	<i>0.98(1)</i>
DP(182.)	1.32	2.37	<i>1.00</i>	2.45	5.97	1.42(6)	1.12(4)	1.13(5)	1.03(2)	1.06(3)	<i>0.97(1)</i>
DP(364.)	1.34	2.79	<i>1.00</i>	2.84	7.28	1.52(6)	1.15(4)	1.19(5)	1.07(2)	1.11(3)	<i>1.02(1)</i>
Average	1.188	1.475	1.605	1.441	3.085	1.170(6)	1.079(4)	1.100(5)	1.060(3)	1.056(2)	<i>1.029(1)</i>
Variance	0.044	0.351	0.582	0.308	3.607	0.020(5)	<i>0.003(1)</i>	0.06(5)	<i>0.003(1)</i>	<i>0.003(1)</i>	<i>0.003(1)</i>

^a The RMSE are normalized with respect to the most accurate individual surrogate. And the most accurate individual surrogates and ensemble models are italicized.

Table 7 Comparison of the normalized RMSE of individual surrogate and ensemble of surrogates for the all mathematical test problems with four individual surrogates

RMSE ^a	Individual surrogate				Ensemble of surrogates					
	RBF	KRG0	PRS	KRG1	EG	Od	SP	EA	EB	OP
BH(12.)	1.24	<i>1.00</i>	1.58	<i>1.00</i>	<i>1.05(1)</i>	<i>1.05(1)</i>	1.09(5)	1.09(5)	1.06(4)	<i>1.05(1)</i>
BH(24.)	<i>1.00</i>	1.41	1.78	1.45	0.99(2)	1.00(5)	0.99(2)	1.00(5)	<i>0.98(1)</i>	0.99(2)
BH(36.)	<i>1.00</i>	1.58	2.29	1.58	0.99(5)	0.98(4)	0.99(5)	0.97(2)	<i>0.96(1)</i>	0.97(2)
Ca(12.)	1.01	1.25	<i>1.00</i>	1.28	1.06(6)	1.05(3)	1.05(3)	1.05(3)	<i>1.04(1)</i>	<i>1.04(1)</i>
Ca(24.)	<i>1.00</i>	1.56	1.27	1.60	1.12(5)	<i>1.10(1)</i>	1.13(6)	1.11(3)	<i>1.10(1)</i>	1.11(3)
Ca(36.)	1.43	<i>1.00</i>	2.46	1.03	1.08(5)	1.04(2)	1.11(6)	1.07(4)	1.05(3)	<i>1.02(1)</i>
GP(12.)	1.01	<i>1.00</i>	1.14	1.01	<i>0.98(1)</i>	<i>0.98(1)</i>	1.00(5)	1.00(5)	<i>0.98(1)</i>	<i>0.98(1)</i>
GP(24.)	1.27	<i>1.00</i>	2.01	1.02	1.08(5)	1.08(5)	<i>1.04(1)</i>	1.06(4)	<i>1.04(1)</i>	<i>1.04(1)</i>
GP(36.)	1.71	<i>1.00</i>	3.77	1.03	1.20(6)	1.15(5)	1.10(4)	1.09(3)	<i>1.06(1)</i>	<i>1.06(1)</i>
Ha(56.)	1.22	<i>1.00</i>	1.38	1.04	1.02(5)	1.01(3)	1.02(5)	<i>1.00(1)</i>	1.01(3)	<i>1.00(1)</i>
Ha(112.)	1.24	<i>1.00</i>	1.37	1.03	1.02(5)	1.01(2)	1.02(5)	1.01(2)	1.01(2)	<i>1.00(1)</i>
ER(110.)	<i>1.00</i>	2.03	1.07	1.56	1.10(6)	1.03(3)	1.07(5)	1.02(2)	1.03(3)	<i>0.98(1)</i>
ER(220.)	1.01	2.12	<i>1.00</i>	1.60	1.12(6)	1.04(3)	1.09(5)	1.02(2)	1.04(3)	<i>0.99(1)</i>
DP(182.)	1.28	2.31	<i>1.00</i>	2.37	1.27(6)	1.09(4)	1.10(5)	1.01(2)	1.04(3)	<i>0.97(1)</i>
DP(364.)	1.35	2.81	<i>1.00</i>	2.84	1.40(6)	1.15(4)	1.21(5)	1.07(2)	1.11(3)	<i>1.02(1)</i>
Average	1.186	1.471	1.608	1.421	1.099(6)	1.050(4)	1.067(5)	1.038(3)	1.034(2)	<i>1.016(1)</i>
Variance	0.044	0.346	0.591	0.298	0.013(6)	0.003(4)	0.004(5)	<i>0.002(1)</i>	<i>0.002(1)</i>	<i>0.002(1)</i>

^a The RMSE are normalized with respect to the most accurate individual surrogate. And the most accurate individual surrogates and ensemble models are italicized.

Table 8 Comparison of the normalized RMSE of individual surrogate and ensemble of surrogates for the all mathematical test problems with three individual surrogates

RMSE ^a	Individual surrogate			Ensemble of surrogates					
	RBF	KRG0	PRS	EG	Od	SP	EA	EB	OP
BH(12.)	1.24	<i>1.00</i>	1.56	1.10(6)	1.08(2)	1.09(5)	1.08(2)	1.08(2)	<i>1.07(1)</i>
BH(24.)	<i>1.00</i>	1.37	1.76	0.99(4)	0.99(4)	0.98(3)	0.99(4)	<i>0.96(1)</i>	0.97(2)
BH(36.)	<i>1.00</i>	1.55	2.32	1.00(6)	0.97(5)	0.95(3)	0.96(4)	<i>0.92(1)</i>	0.93(2)
Ca(12.)	1.01	1.25	<i>1.00</i>	1.02(2)	<i>1.01(1)</i>	1.04(5)	1.04(5)	1.02(2)	1.02(2)
Ca(24.)	<i>1.00</i>	1.56	1.28	<i>1.07(1)</i>	<i>1.07(1)</i>	1.13(6)	1.11(5)	1.08(3)	1.08(3)
Ca(36.)	1.49	<i>1.00</i>	2.56	1.23(6)	1.13(4)	1.14(5)	1.09(2)	1.09(2)	<i>1.06(1)</i>
GP(12.)	1.02	<i>1.00</i>	1.13	<i>0.99(1)</i>	<i>0.99(1)</i>	1.01(5)	1.01(5)	<i>0.99(1)</i>	<i>0.99(1)</i>
GP(24.)	1.25	<i>1.00</i>	2.00	1.14(6)	1.11(5)	<i>1.05(1)</i>	1.06(2)	1.06(2)	1.06(2)
GP(36.)	1.74	<i>1.00</i>	3.77	1.39(6)	1.24(4)	1.08(3)	<i>1.07(1)</i>	<i>1.07(1)</i>	1.08(3)
Ha(56.)	1.22	<i>1.00</i>	1.37	1.05(6)	1.03(5)	1.02(4)	<i>1.00(1)</i>	1.01(3)	<i>1.00(1)</i>
Ha(112.)	1.25	<i>1.00</i>	1.36	1.04(6)	1.03(5)	1.02(3)	<i>1.01(1)</i>	1.02(3)	<i>1.01(1)</i>
ER(110.)	<i>1.00</i>	2.04	1.08	1.06(6)	1.00(3)	1.04(5)	0.99(2)	1.00(3)	<i>0.97(1)</i>
ER(220.)	1.02	2.13	<i>1.00</i>	1.08(6)	1.01(2)	1.07(5)	1.01(2)	1.02(4)	<i>0.98(1)</i>
DP(182.)	1.30	2.34	<i>1.00</i>	1.14(6)	1.04(4)	1.08(5)	1.01(2)	1.03(3)	<i>0.98(1)</i>
DP(364.)	1.34	2.82	<i>1.00</i>	1.22(6)	1.08(3)	1.16(5)	1.05(2)	1.08(3)	<i>1.02(1)</i>
Average	1.192	1.471	1.613	1.100(6)	1.052(4)	1.057(5)	1.030(3)	1.028(2)	<i>1.015(1)</i>
Variance	0.048	0.355	0.605	0.012(6)	0.005(5)	0.003(4)	<i>0.002(1)</i>	<i>0.002(1)</i>	<i>0.002(1)</i>

^a The RMSE are normalized with respect to the most accurate individual surrogate. And the most accurate individual surrogates and ensemble models are italicized

Table 9 Comparison of the normalized RMSE of individual surrogate and ensemble of surrogates for the all engineering test problems with five individual surrogates

RMSE ^a	Individual surrogate					Ensemble of surrogates					
	RBF	KRG0	PRS	KRG1	GP	EG	Od	SP	EA	EB	OP
Ibeam (30.)	1.00	1.75	1.44	1.45	1.12	1.08(6)	1.06(5)	1.05(4)	1.04(2)	1.04(2)	<i>1.01(1)</i>
Fortini (30.)	1.00	106.61	6.15	15.47	1.17	2.42(6)	1.86(5)	1.08(2)	1.09(4)	1.08(2)	<i>1.03(1)</i>
Nozzle flow rate (12.)	5.64	44.78	1.00	1.07	9.81	1.93(6)	1.34(5)	<i>0.82(1)</i>	0.87(4)	0.84(3)	<i>0.82(1)</i>
Nozzle thrust (12.)	1.13	5.61	2.42	1.00	1.33	1.17(6)	1.12(5)	1.04(4)	1.01(3)	<i>0.98(1)</i>	<i>0.98(1)</i>
VCE thrust (100.)	1.21	1.71	2.17	1.00	1.43	1.03(6)	1.01(5)	1.00(4)	0.97(2)	0.99(3)	<i>0.96(1)</i>
VCE sfc (100.)	1.55	1.55	2.19	1.00	2.43	1.34(6)	1.19(4)	1.24(5)	1.11(2)	1.15(3)	<i>1.08(1)</i>

^a The RMSE are normalized with respect to the most accurate individual surrogate. And the most accurate individual surrogates and ensemble models are italicized

Table 10 The computational time among different ensemble models for all mathematical test problems with five individual surrogates

Computational time (s)	EG	Od	SP	EA	EB	OP
BH(12.)	0.029(1.00)	0.029(1.00)	0.029(1.00)	0.029(1.00)	0.029(1.00)	0.036(1.24)
BH(24.)	0.067(1.00)	0.068(1.01)	0.067(1.00)	0.068(1.01)	0.068(1.01)	0.093(1.39)
BH(36.)	0.119(1.00)	0.119(1.00)	0.120(1.01)	0.120(1.01)	0.120(1.01)	0.177(1.49)
Ca(12.)	0.029(1.00)	0.029(1.00)	0.029(1.00)	0.029(1.00)	0.029(1.00)	0.036(1.24)
Ca(24.)	0.067(1.00)	0.067(1.00)	0.067(1.00)	0.067(1.00)	0.067(1.00)	0.093(1.39)
Ca(36.)	0.117(1.00)	0.117(1.00)	0.117(1.00)	0.117(1.00)	0.117(1.00)	0.174(1.49)
GP(12.)	0.029(1.04)	0.029(1.04)	0.028(1.00)	0.029(1.04)	0.029(1.04)	0.035(1.25)
GP(24.)	0.067(1.00)	0.067(1.00)	0.067(1.00)	0.067(1.00)	0.067(1.00)	0.092(1.37)
GP(36.)	0.116(1.00)	0.116(1.00)	0.116(1.00)	0.116(1.00)	0.117(1.01)	0.172(1.48)
Ha(56.)	0.259(1.00)	0.259(1.00)	0.259(1.00)	0.259(1.00)	0.258(1.00)	0.389(1.51)
Ha(112.)	0.918(1.00)	0.919(1.00)	0.918(1.00)	0.918(1.00)	0.918(1.00)	1.437(1.57)
ER(110.)	1.029(1.00)	1.028(1.00)	1.026(1.00)	1.030(1.00)	1.028(1.00)	1.521(1.48)
ER(220.)	5.257(1.00)	5.265(1.00)	5.267(1.00)	5.262(1.00)	5.267(1.00)	7.229(1.38)
DP(182.)	4.388(1.00)	4.388(1.00)	4.387(1.00)	4.386(1.00)	4.386(1.00)	5.755(1.31)
DP(364.)	36.26(1.00)	36.28(1.00)	36.28(1.00)	36.32(1.00)	36.29(1.00)	41.73(1.15)

the ensemble models considered in this paper increases. And the computational time of computing the distance between each observed point also increases. For the high-dimensional test problems, the increasing computation time to compute the cross-validation error is more than that to compute the distance between each observed point. This is the reason why the normalized computational time decreases as the number of points in the training set increases for ExtendedRosen and DixonPrice test problems.

7 Conclusions

In this work, we propose an optimal pointwise weighted ensemble (OPWE) method, wherein the optimal pointwise weight factors are obtained based on minimization of the local mean square error which is constructed by the global-local error (GLE). Via six well-known mathematical functions with varying dimensions and numbers of training points and four engineering problems, it is proved that the proposed OPWE is better than the other ensembles of surrogates in terms of both accuracy and robustness. GLE is a more accurate surrogate than the cross-validation error for the local error at the observed points. The method proposed in this paper can be utilized to prevent the bias in error estimation of GLE for local error and to improve the accuracy and robustness of ensemble models.

Funding information This study received financial support from the National Natural Science Foundation of China (Nos. 51876176 and 51576163) and MIIT of China.

Compliance with ethical standards

Conflict of interests The authors declare that they have no conflict of interest.

Replication of results The method proposed in this paper is implemented in the surrogates module of SURROGATES (2011), a MATLAB-based toolbox for multidimensional function approximation and optimization methods. The results presented in this paper can thus be reproduced easily.

References

- Acar E, Rais-Rohani M (2009) Ensemble of metamodels with optimized weight factors. *Struct Multidisc Optim* 37:279–294. <https://doi.org/10.1007/s00158-008-0230-y>
- Acar E (2010) Various approaches for constructing an ensemble metamodels using local measure. *Struct Multidisc Optim* 42:879–896. <https://doi.org/10.1007/s00158-010-0520-z>
- Chen L, Qiu H, Jiang C, Cai X, Gao L (2018) Ensemble of surrogates with hybrid method using global and local measures for engineering design. *Struct Multidisc Optim* 57:1711–1729. <https://doi.org/10.1007/s00158-017-1841-y>

- Ferreira WG, Serpa AL (2016) Ensemble of metamodels: the augmented least squares approach. *Struct Multidisc Optim* 53:1019–1046. <https://doi.org/10.1007/s00158-015-1366-1>
- Forrester AIJ, Keane AJ (2009) Recent advances in surrogate based optimization. *Prog Aerosp Sci* 45:50–79
- Goel T, Haftka RT, Shyy W, Queipo NV (2007) Ensemble of surrogates. *Struct Multidisc Optim* 33:199–216. <https://doi.org/10.1007/s00158-006-0051-9>
- Gunn SR (1997) Support vector machines for classification and regression. Technical Report, University of Southampton
- Hardy RL (1971) Multiquadric equations of topography and other irregular surfaces. *J Geophys Res* 76:1950–1915. <https://doi.org/10.1029/JB076i008p01905>
- Imani M, Ghoreishi SF, Braga-Neto UM (2017) Maximum-likelihood adaptive filter for partially observed boolean dynamical systems. *IEEE Trans Signal Process* 65:357–371
- Imani M, Ghoreishi SF, Braga-Neto UM (2018) Bayesian control of large MDPs with unknown dynamics in data-poor environments. *Advances in Neural Information Processing Systems*: 8146–8156
- Imani M, Ghoreishi SF, Allaire D, Braga-Neto UM (2018) MFBO-SSM: Multi-fidelity bayesian optimization for fast inference in state-space models. *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)* 33:7858–7865
- Jekabsons G (2009) RBF: Radial basis function interpolation for matlab/octave. Riga Technical University, Latvia. version 1.1 ed
- Jin R, Chen W, Simpson TW (2001) Comparative studies of metamodeling techniques under multiple modeling criteria. *Struct Multidisc Optim* 23:1–13. <https://doi.org/10.1007/s00158-001-0160-4>
- Lee SH, Kwak BM (2006) Response surface augmented moment method for efficient reliability analysis. *Struct Saf* 28:261–272
- Lee Y, Choi DH (2014) Pointwise ensemble of meta-models using v nearest points cross-validation. *Struct Multidisc Optim* 50:383–394. <https://doi.org/10.1007/s00158-014-1067-1>
- Li G, Aute V, Azarm S (2010) An accumulative error based adaptive design of experiments for offline metamodeling. *Struct Multidisc Optim* 40:137–155. <https://doi.org/10.1007/s00158-009-0395-z>
- Liu H, Xu S, Wang X (2016) Optimal weighted pointwise ensemble of radial basis functions with different basis functions. *AIAA J* 54:3117–3133. <https://doi.org/10.2514/1.J054664>
- Lophaven SN, Nielsen HB, Sondergaard J (2002) DACE - a matlab kriging toolbox. Tech. Rep. IMM-TR-2002-12. Technical University of Denmark
- MacKay DJC (1998) Introduction to Gaussian processes. In: Bishop CM (ed) *Neural networks and machine learning*, vol 168 of NATO ASI Series, 1998. Springer, Berlin, pp 135–165. <https://doi.org/10.1007/978-0-387-21681-2-8>
- Messac A, Mullur AA (2008) A computationally efficient metamodeling approach for expensive multiobjective optimization. *Optim Eng* 9:37–67
- Meng Z, Zhang D, Li G, Yu B (2019) An importance learning method for non-probabilistic reliability analysis and optimization. *Struct Multidisc Optim* 59:1255–1271. <https://doi.org/10.1007/s00158-018-2128-7>
- 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
- Rasmussen CE, Williams CK (2006) *Gaussian processes for machine learning*. The MIT Press
- Sacks J, Welch WJ, Mitchell TJ, Wynn HP (1989) Design and analysis of computer experiments. *Stat Sci* 4(4):409–435. <https://doi.org/10.1214/ss/1177012413>

- Simpson TW, Peplinski JD, Koch PN, Allen JK (2001) Metamodels for computer-based engineering design: survey and recommendations. *Eng Comput* 17:129–150. <https://doi.org/10.1007/PL00007198>
- Smith M (1993) *Neural networks for statistical modeling*. Von Nostrand Reinhold, New York
- Viana FAC, Haftka RT, Steffen V (2009) Multiple surrogates: how cross-validation errors help us to obtain the best predictor. *Struct Multidisc Optim* 39:439–457. <https://doi.org/10.1007/s00158-008-0338-0>
- Viana FAC (2011) SURROGATES toolbox user's guide version 3.0. <http://sites.google.com/site/felipeacviana/surrogate-stoolbox>
- Wang GG, Shan S (2007) Review of metamodeling techniques in support of engineering design optimization. *ASME J Mech Des* 129(4):370–380
- Yin H, Fang H, Wen G, Gutowski M, Xiao Y (2018) On the ensemble of metamodels with multiple regional optimized weight factors. *Struct Multidisc Optim* 58:245–263. <https://doi.org/10.1007/s00158-017-1891-1>
- Zhang X, Wang Z, Zhou L, Liu Z (2016) Multidisciplinary design optimization on conceptual design of aero-engine. *International Journal of Turbo and Jet-Engine* 33(2):195–208. <https://doi.org/10.1515/tjj-2015-0024>
- Zhou XJ, Ma YZ, Li XF (2011) Ensemble of surrogates with recursive arithmetic average. *Struct Multidisc Optim* 44:651–671. <https://doi.org/10.1007/s00158-011-0655-6>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.