Complex & Intelligent Systems (2022) 8:1015–1034
https://doi.org/10.1007/s40747-021-00568-7

ORIGINAL ARTICLE

Polynomial Response Surface based on basis function selection by multitask optimization and ensemble modeling

Yong Zhao[1] · Siyu Ye[1] · Xianqi Chen[1] · Yufeng Xia[1] · Xiaohu Zheng[1]

Received: 14 July 2021 / Accepted: 19 October 2021 / Published online: 10 November 2021

Abstract
Polynomial Regression Surface (PRS) is a commonly used surrogate model for its simplicity, good interpretability, and computational efficiency. The performance of PRS is largely dependent on its basis functions. With limited samples, how to correctly select basis functions remains a challenging problem. To improve prediction accuracy, a PRS modeling approach based on multitask optimization and ensemble modeling (PRS-MOEM) is proposed for rational basis function selection with robustness. First, the training set is partitioned into multiple subsets by the cross validation method, and for each subset a sub-model is independently constructed by optimization. To effectively solve these multiple optimization tasks, an improved evolutionary algorithm with transfer migration is developed, which can enhance the optimization efficiency and robustness by useful information exchange between these similar optimization tasks. Second, a novel ensemble method is proposed to integrate the multiple sub-models into the final model. The significance of each basis function is scored according to the error estimation of the sub-models and the occurrence frequency of the basis functions in all the sub-models. Then the basis functions are ranked and selected based on the bias-corrected Akaike's information criterion. PRS-MOEM can effectively mitigate the negative influence from the sub-models with large prediction error, and alleviate the uncertain impact resulting from the randomness of training subsets. Thus the basis function selection accuracy and robustness can be enhanced. Seven numerical examples and an engineering problem are utilized to test and verify the effectiveness of PRS-MOEM.

Keywords Polynomial Response Surface · Basis function selection · Multitask optimization · Ensemble modeling

Introduction

Despite the tremendous promotion in computer processing power, the computationally expensive problem occurs frequently in multiple scientific and engineering disciplines where complex computer simulations are used. In these cases, obtaining more data means additional experiments and thus it results in highly non-trivial computational expense (Forrester and Keane [1]). As a result, surrogate models have been widely used to replace the complex simulation models (Namura et al. [2]). Their application fields involve multidisciplinary design optimization (Yao et al. [3]), uncertainty analysis (Yao et al. [4]), and so on.

At present, the commonly used surrogate models are: Polynomial Response Surface (PRS) (Goel et al. [5]), Multivariate Adaptive Regression Splines (MARS) (Gu and Wahba [6]), Kriging (Clark et al. [7]), Radial Basis Function (RBF) (Yao et al. [8]), Support Vector Regression (SVR) (Clarke et al. [9]). There are also some hybrid surrogate modeling paradigms where different surrogate models are combined to offer effective solutions ((Zhang et al. [10])(Yin et al. [11]). Among these surrogates, PRS is a popular surrogate model because of its simplicity, good interpretability, and computational efficiency (Bhosekar and Ierapetritou [12]). However, PRS is unsuitable for the non-linear, multimodal, multi-dimensional design landscapes (Forrester and

✉ Siyu Ye
yesiyu13gfkd@163.com

Yong Zhao
zhaoyong@nudt.edu.cn

Xianqi Chen
chenxianqi12@nudt.edu.cn

Yufeng Xia
xiayufeng15@outlook.com

Xiaohu Zheng
zhengboy320@163.com

[1] College of Aerospace Science and Engineering, National University of Defense Technology, Changsha 410073, Hunan, China

Keane [1]). Because of the limitations existing in PRS, some enhanced versions which include removing or suppressing unnecessary variables are proposed, such as subset selection (Furnival and Wilson [13]) or regularization (Tibshirani [14]). Subset selection methods address the trade-off between prediction error and the regression model complexity by selecting a subset of variables. Stepwise regression is an effective way to solve the subset selection problem (Hosseinpour et al. [15]). Borrowing the idea of stepwise regression, Giustolisi et al. (O. Giustolisi and Doglioni [16]) propose Evolutionary Polynomial Regression (EPR), which uses an evolutionary process rather than follows the hill-climbing method of stepwise regression. For the regularization method, Least absolute shrinkage and selection (Lasso) is widely used (Tibshirani [14]), which shrinks some coefficients and sets others to zero to retain the good features of both subset selection and ridge regression. Zou et al. (Zou and Hastie [17]) propose the elastic net (EN) technique to fix the problem of failing to select correlated grouped variables that existed in the Lasso.

Either for subset selection or regularization methods, their performance is largely dependent on the basis functions included in the model. By choosing the significant terms, the prediction ability of the model would be enhanced. Commonly, the basis function selection is based on a specific error estimation of the model. An error estimation method that accords with the true error of the model could guide the selection of basis functions more effectively. Currently, the mean squared error estimation method based on cross validation (CV), such as prediction error sum of squares (PRESS) (Goel et al. [18]), is a popular way to estimate the global accuracy of the model. However, with limited samples available, the mean squared error estimation method based on cross validation is influenced by the data partition scheme, which may not estimate the true global average error well. Thus, in this situation, it is difficult to select significant basis functions effectively for PRS. To avoid the problem, Gu et al. (Gu and Wei [19]) propose a robust model structure selection method, which could select the significant model terms according to the overall mean absolute error of the resampled subsets. Although the method achieves good results in some numerical examples, it leads to large computational costs. Besides, the aforementioned modeling methods are based on the error estimation of all the CV subsets, which may result in incorrect basis function selection due to significant influence from certain subsets with large estimation error (as discussed in Sect. 2.3).

In this paper, a novel approach based on multitask optimization and ensemble modeling (PRS-MOEM) is proposed to effectively mitigate the negative influence of the subsets with large estimation error due to the random partition, and enhance the basis function selection accuracy and robustness. First, multiple subsets are partitioned from the training set based on cross validation. Instead of the traditional modeling method which directly builds a single model guided by the error estimation based on cross validation, multiple sub-models are constructed by building the surrogate for each subset. The multiple sub-model modeling processes are solved in parallel by multiple optimization tasks. To improve the optimization performance, multitask optimization can be adopted (Ong and Gupta [20]; Naik and Rangwala [21]), and an improved evolutionary algorithm with transfer migration is developed to solve multitask optimization problem, which can significantly enhance the optimization efficiency and robustness by useful information exchange between the similar optimization tasks. Second, a novel ensemble method is proposed to integrate the multiple sub-models into the final optimal one. Actually, there are relevant researches on ensemble modeling which integrates multiple models into one model. However, previous studies are usually conducted by the weighted sum approach (Fang et al. [22]; Zhou and Jiang [23]). Since some sub-models may deviate from the true model greatly due to the specific training subset features (the subsets are randomly partitioned), this weighted sum approach may result in wrongly selecting basis functions. Thus, the interpretability of the PRS model would be reduced as well as its accuracy. To obtain a well performed ensemble, in this paper a scoring method is proposed to measure the significance of each basis function according to the error estimation of the sub-models and the occurrence frequency of these basis functions in all the sub-models. The basis functions are ranked according to the significance scores in descending order. Each time add a basis function into the ensemble and measure the model accuracy by the bias-corrected Akaike's information criterion (AICc). The ensemble with the lowest AICc is chosen as the final model. In this way, the negative influence from the sub-models with large prediction error can be mitigated, and the uncertain impact resulting from the random partition of subsets can be alleviated. Thus the basis function selection accuracy as well as algorithm robustness can be effectively enhanced. The main contributions of this paper can be summarized as follows:

- A PRS modeling problem is decomposed into multiple subproblems, which are to build the sub-model for each subset separately. Therefore, the potential of each subset can be fully explored and the phenomenon of wrong dominance by specific subsets can be mitigated.
- Multitask optimization is introduced to simultaneously solve the subproblems, and an improved evolutionary algorithm with transfer migration is developed to enhance the optimization efficiency and robustness.
- An ensemble modeling method for integrating multiple sub-models is proposed. Using a novel scoring strategy, the method sorts each basis function of the sub-models,

combines them serially, and selects a final optimum model.

The rest of the paper is organized as follows. In Sect. 2, a brief review of Polynomial Response Surface (PRS) is introduced and the disadvantage of the traditional modeling method based on CV is analyzed. In Sect. 3, the PRS method based on Basis Function Selection by Multitask Optimization and Ensemble Modeling (PRS-MOEM) is developed in detail. In Sect. 4, the proposed PRS-MOEM is testified with seven numerical examples and one practical engineering problem, followed by conclusions in the final section.

Preliminary

Polynomial regression surface

The PRS model is derived from the linear regression model, the matrix form can be written as

$$\hat{y}(\boldsymbol{x}) = [f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_{nvars}(\boldsymbol{x})] \cdot \boldsymbol{\beta} \tag{1}$$

where $\boldsymbol{x} = [x_1, x_2, ..., x_m]^T$ is a sampled point. m is the number of variables. $\boldsymbol{\beta} = [\beta_1, \beta_2, ..., \beta_{nvars}]^T$ is the regression coefficients vector. $[f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), ..., f_{nvars}(\boldsymbol{x})]$ is the basis function vector, and f_i is a basis function. $nvars$ is the number of basis functions. Given a set of training points $\boldsymbol{x}^{(l)} \in \mathbb{R}^m, l = 1, 2, ..., n$, and the corresponding actual response vector $\boldsymbol{y} = [y^{(1)}, y^{(2)}, ..., y^{(n)}]^T$, the design matrix is defined as

$$\mathbf{F} = \begin{bmatrix} f_1(\boldsymbol{x}^{(1)}) & \cdots & f_{nvars}(\boldsymbol{x}^{(1)}) \\ \cdots & \cdots & \cdots \\ f_1(\boldsymbol{x}^{(n)}) & \cdots & f_{nvars}(\boldsymbol{x}^{(n)}) \end{bmatrix} \tag{2}$$

The least squares method is often used to solve the regression coefficients as

$$\boldsymbol{\beta} = \mathbf{F}^+ \cdot \boldsymbol{y} \tag{3}$$

where $\mathbf{F}^+ = (\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T$ is the Moore-Penrose pseudo-inverse of $\mathbf{F}$.

To lower the mutual coherence of the design matrix, the multivariable Legendre orthogonal polynomial (Fan et al. [24]) is applied to form the basis function in this paper, which is

$$f_i(\boldsymbol{x}) = \prod_{j=1}^{m} L_{\eta_j}\left(x_j\right) \tag{4}$$

where η_j is the order of the jth univariate Legendre polynomial $L_{\eta_j}\left(x_j\right)$, and $\sum_{j=1}^{m} \eta_j = P$, P is a user-defined highest order of polynomials. $L_{\eta_j}\left(x_j\right)$ is determined by the recursive definition. It is supposed that $L_0\left(x_j\right) = 1$ and $L_1\left(x_j\right) = x_j$, then

$$\left(\eta_j + 1\right) L_{\eta_j+1}\left(x_j\right) = \left(2\eta_j + 1\right) x_j L_{\eta_j}\left(x_j\right) - \eta_j L_{\eta_j-1}\left(x_j\right) \tag{5}$$

Traditional PRS modeling method based on cross validation

PRS modeling first should define the basis function vector, which needs separate sample set for model validation and basis function selection. However, with limited samples available, it is difficult to obtain separate sample set for surrogate validation. To address this problem, the cross-validation method is widely used as it can provide good error estimation when the sample size is small (Bischl et al. [25]). For K-fold cross-validation, the training set $D = \{(\boldsymbol{x}^{(l)}, \boldsymbol{y}^{(l)}), l = 1, 2, ..., n\}$ is randomly partitioned into K disjoint sets of approximately equal size, denoted as $D_1, D_2, ..., D_K$. Note that for small data size, 10-fold cross-validation provides almost unbiased estimate of prediction error. With a relatively large number of samples, small K is preferred, such as 5, to avoid high computational cost (Simon [26]). For $k = 1, 2, ..., K$, D_k is the validation subset with the corresponding training subset $D^{(-k)} = D - D_k$. The candidate basis function set is defined as $\mathbf{\Phi} = \{f_i\}_{i=1,...,nvars}$. The tradition modeling method tries to find an active basis function set from $\mathbf{\Phi}$ by minimizing the error estimation of the final model, where . Here 'active' means that the basis function is selected into the final model, while 'inactive' is opposite. First, the selected active basis function vector with N_{active} items is defined as

$$\begin{aligned} &\mathbf{S} = [f_{a1}, f_{a2}, ..., f_{aN_{active}}]^T \\ &f_{ai} \in \mathbf{\Phi}, 1 \le i \le N_{active} \end{aligned} \tag{6}$$

The design matrix $\mathbf{F}^{(-k)}$ is constructed with the active basis function vector $\mathbf{S}$ for $D^{(-k)}$ as

$$\begin{aligned} &\mathbf{F}^{(-k)} = \left[\mathbf{S}(\boldsymbol{x}^{k1}), \mathbf{S}(\boldsymbol{x}^{k2}), ..., \mathbf{S}(\boldsymbol{x}^{kn_k})\right]^T \\ &\boldsymbol{x}^{kj} \in D^{(-k)}, 1 \le j \le n_k \end{aligned} \tag{7}$$

where the superscript T means the transverse of the matrix, n_k is the number of points in the subset $D^{(-k)}$. Then the regression coefficient vector $\boldsymbol{\beta}^{(-k)}$ is calculated by

$$\boldsymbol{\beta}^{(-k)} = (\mathbf{F}^{(-k)})^+ \cdot \boldsymbol{y}^{(-k)} \tag{8}$$

where $\boldsymbol{y}^{(-k)}$ is the actual response vector of the training samples in $D^{(-k)}$, and $(\mathbf{F}^{(-k)})^+$ is the Moore-Penrose pseudo-inverse of $\mathbf{F}^{(-k)}$. Based on $\mathbf{S}$ and $\boldsymbol{\beta}^{(-k)}$, the sub-model corresponding to the training subset $D^{(-k)}$ can be obtained as

$$\hat{y}_k(\boldsymbol{x}) = \mathbf{S}(\boldsymbol{x}) \cdot \boldsymbol{\beta}^{(-k)} \tag{9}$$

Based on D_k, the prediction error of the model can be estimated. As the commonly used ordinary cross-validation error estimation criterion (CV) would lead to large bias when the sample size is small (Yanagihara et al. [27]), the bias-corrected cross-validation criterion (CCV) is chosen in this paper for error estimation, which is

$$\begin{aligned} \boldsymbol{\Theta}(\boldsymbol{\beta}^{(-k)}) = \frac{1}{n-n_k} \cdot \sum_{i=1}^{n-n_k} (|y^{(i)} - \hat{y}_k(\boldsymbol{x}^{(i)})|_{\boldsymbol{x}^{(i)} \in D_k}) \\ + \frac{1}{n_k} \cdot \sum_{j=1}^{n_k} (\tfrac{1}{2n}|y^{(j)} - \hat{y}_k(\boldsymbol{x}^{(j)})|_{\boldsymbol{x}^{(j)} \in D^{(-k)}}) \end{aligned} \tag{10}$$

CCV calculates the prediction error of the model in the first term, and considers the fitting error meanwhile in the second term, which could avoid large bias with limited samples. For $k = 1, 2, ..., K$, repeat the aforementioned steps and obtain the prediction errors of all the sub-models based on the active basis function vector **S**. By minimizing the error sum $\sum_{k=1}^{K} \boldsymbol{\Theta}(\boldsymbol{\beta}^{(-k)})$, the optimal active basis function vector can be obtained. The diagram of the above method is shown in Fig. 1a, and the optimization task can be formulated as

$$\begin{aligned} &\text{Find} \quad \mathbf{S} = [f_{a1}, f_{a2}, ..., f_{aN_{active}}]^T,\ f_{ai} \in \boldsymbol{\Phi} \\ &\text{Min} \quad \sum_{k=1}^{K} \boldsymbol{\Theta}(\boldsymbol{\beta}^{(-k)}) \\ &\text{s.t.} \quad \boldsymbol{\beta}^{(-k)} = (\mathbf{F}^{(-k)})^{+} \cdot \boldsymbol{y}^{(-k)} \\ &\qquad \mathbf{F}^{(-k)} = \left[\mathbf{S}(\boldsymbol{x}^{k1}), \mathbf{S}(\boldsymbol{x}^{k2}), ..., \mathbf{S}(\boldsymbol{x}^{kn_k})\right]^T \\ &\qquad \boldsymbol{x}^{kj} \in D^{(-k)}, 1 \le j \le n_k, 1 \le k \le K \\ &\qquad N_{active} \le \min_{1 \le k \le K} (n_k) - 1 \end{aligned} \tag{11}$$

The number of active basis functions in the final model is N_{active} and has to meet the constraint $N_{active} \le \min\limits_{1 \le k \le K} (n_k) - 1$ (Lee et al. [28]).

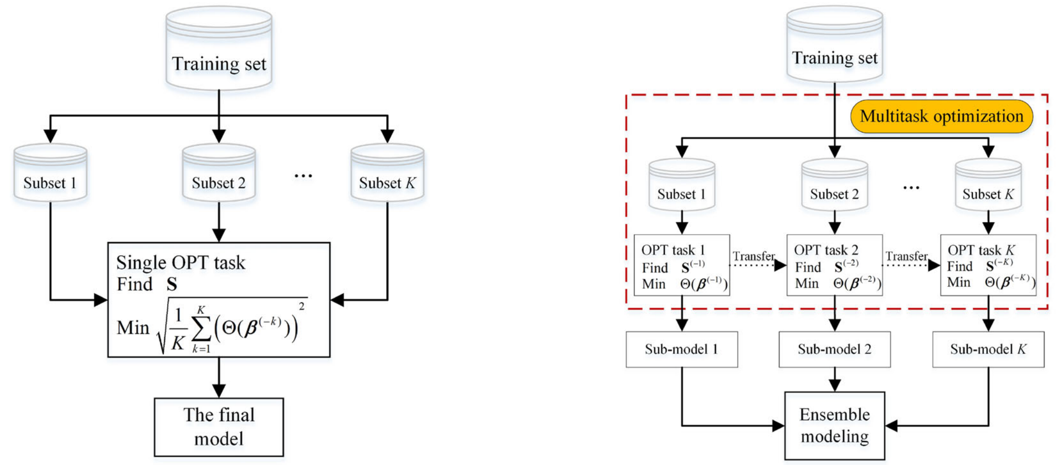
Disadvantage of the traditional PRS modeling based on cross validation

In solving Eq.(11), the optimal active basis function vector **S** is obtained by minimizing the total sub-model error sum $\sum_{k=1}^{K} \boldsymbol{\Theta}(\boldsymbol{\beta}^{(-k)})$, which may easily lead to incorrect selection due to the significant influence from certain sub-models with large error estimation. A one-dimensional problem is used for illustration and presented in Fig. 2. The sample set is divided into two subsets, labeled as subset A (blue dots) and subset B (red dots) respectively. To demonstrate the influence of the subsets on the final model construction, build the optimal sub-models for each training subset first. With the training subset A and the validation subset B, the sub-model 1 (brown dash line) is obtained by minimizing the subset estimation error stated in Eq.(10). And similarly, the sub-model 2 (red dash line) is obtained with the training subset B and the validation subset A. Obviously, the nonlinear sub-model 1 is more in line with the true model (green dash line), and the sub-model 2 is just a linear model (only linear basis functions are selected) which deviates a lot from the true response. This indicates that with improper subset partition (which may happen with large probability due to the random CV partition method), the optimal model obtained by minimizing the estimation error is far from the true model, which means it fails to correctly identify the basis functions with training subset B.

Then by solving Eq.(11) to minimize the total error sum of all the subsets, the final model is obtained (black solid line). It can be observed that the final model is also a linear model, which is very close to sub-model 2. It is because the large error estimation of sub-model 2 dominated the total error sum, which accordingly guides the optimization search towards minimizing the error estimation of sub-model 2. Therefore, the basis function selection of the final model is close to the sub-model 2. The underlying reason for this problem is that the active basis function vector is optimized by considering its performance on all the subsets (the total error sum). Then the optimization process can be easily dominated by some specific subsets for which the error estimation is large. To address this problem, an intuitive idea is to optimize the active basis function vector for each subset separately, so that the potential of each subset can be fully explored and the phenomenon of wrong dominance by specific subsets can be mitigated. Then based on all the potential active basis function vectors, the final model can be ensemble according to a certain criterion. Based on this idea, in this paper a novel PRS modeling approach is proposed. It optimizes the active basis function vector and constructs the corresponding sub-model for each subset separately and in parallel based on multi-task optimization (MO). Then the active basis functions are scored and selected by ensemble modeling (EM). This PRS modeling framework is developed in Sect. 3.

Multitask optimization

MO (Jin et al. [29]; Liao et al [30]) can simultaneously solves multiple tasks in a single run and achieve better performance with positive knowledge transfer, which has shown high efficiency on expensive optimization problem (Ding et al. [31]; Wang et al. [32]). As discussed in the above section, a PRS modeling optimization problem can be decomposed into K subproblems which are optimizing active basis func-


Training set
Subset 1
Subset 2
...
Subset K
Single OPT task
Find S
Min $\sqrt{\frac{1}{K}\sum_{k=1}^{K}\left(\Theta(\beta^{(-k)})\right)^2}$
The final model
Training set
Multitask optimization
Subset 1
Subset 2
...
Subset K
OPT task 1 Find $S^{(-1)}$ Min $\Theta(\beta^{(-1)})$
Transfer
OPT task 2 Find $S^{(-2)}$ Min $\Theta(\beta^{(-2)})$
Transfer
OPT task K Find $S^{(-K)}$ Min $\Theta(\beta^{(-K)})$
Sub-model 1
Sub-model 2
Sub-model K
Ensemble modeling

(a) The traditional PRS modeling method (b) Multitask optimization and ensemble modeling method

Fig. 1 The diagram of PRS modeling framework

tion vector for each subset separately. Due to the similarity of these subproblems and the independence of optimization processes, MO is introduced to construct the sub-model for each subset in this paper. Suppose that $\mathbf{S}^{(-k)}$ is feasible active basis function vector of the kth optimization task $\mathbf{\Theta}_k$, then the subproblems are to be simultaneously addressed can be formulated as

$$\left\{\mathbf{S}^{(-1)}, \mathbf{S}^{(-2)}, ..., \mathbf{S}^{(-K)}\right\} = \{\arg\min \mathbf{\Theta}_1, \arg\min \mathbf{\Theta}_2, ..., \arg\min \mathbf{\Theta}_K\} \quad (12)$$

The details of the optimization problem are clarified in the next section.

PRS-MOEM method

The PRS-MOEM framework is shown in Fig. 1b. It mainly includes two key parts, namely the multitask optimization to obtain potential active basis function vector and construct optimal sub-models for all the subsets based on K-fold cross validation, and the ensemble modeling to integrate all the sub-models into the final model. Furthermore, to effectively solve the multitask optimization problem, an improved evolutionary algorithm with transfer migration is developed.

Multitask optimization for sub-model construction

For the subset D_k and $D^{(-k)}$, the active basis function vector can be obtained by solving the following optimization problem:

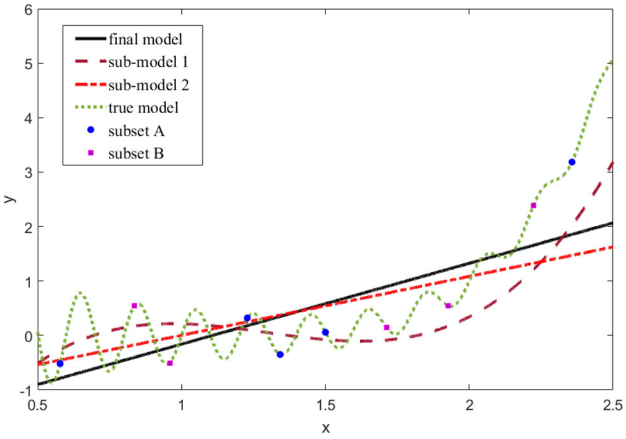


Fig. 2 The illustration of PRS modeling being greatly influenced by specific subsets

$$\begin{aligned}
&\text{Find} \quad \mathbf{S}^{(-k)} = [f_{a1}, f_{a2}, ..., f_{aN_{active}^{(-k)}}]^T, f_{ai} \in \mathbf{\Phi} \\
&\text{Min} \quad \mathbf{\Theta}(\boldsymbol{\beta}^{(-k)}) \\
&\text{s.t.} \quad \boldsymbol{\beta}^{(-k)} = (\mathbf{F}^{(-k)})^{+} \cdot \boldsymbol{y}^{(-k)} \\
&\qquad \mathbf{F}^{(-k)} = \left[\mathbf{S}(\boldsymbol{x}^{k1}), \mathbf{S}(\boldsymbol{x}^{k2}), ..., \mathbf{S}(\boldsymbol{x}^{kn_k})\right]^T \\
&\qquad \boldsymbol{x}^{kj} \in D^{(-k)}, 1 \le j \le n_k \\
&\qquad N_{active}^{(-k)} \le n_k - 1
\end{aligned} \quad (13)$$

where $\mathbf{S}^{(-k)}$ is the active basis function vector of the kth sub-model trained by the subset $D^{(-k)}$, and $N_{active}^{(-k)}$ is the number of items in this vector.

For the K subsets, K optimization tasks have to be conducted to construct all the sub-models. Considering that the optimization processes of the K sub-models are independent of each other, multitask optimization can be adopted to solve the K optimization tasks in parallel. Meanwhile, the optimization tasks are all for the same purpose as to obtain the optimal surrogate based on the training sample subsets obtained from the same true model. Thus these optimization tasks have an inherent similarity. Previous research has shown that positive transfer can outweigh the deleterious negative transfer especially when there are some prior understandings of the similarity between the black-box optimization tasks (Cheng et al. [33]; Feng et al. [34]). Thus, herein the transfer migration method is proposed to exchange the useful information between the different optimization tasks to enhance optimization efficiency, and the details are presented in Sect. 3.3.

Another important issue in the optimization problem Eq. (13) is the design space, i.e. the optional basis function set $\boldsymbol{\Phi}$ herein. In this paper, the multivariable Legendre orthogonal polynomials are used to form the basis function set, the items of which are defined by the highest order (denoted as P) of the polynomials. Theoretically, higher order is preferred with larger design space to cover more possible situations. However, with the quickly enlarged $\boldsymbol{\Phi}$ due to the increase of P, the optimization difficulty will increase dramatically due to the fast increase of the optimization variable number. With a long history of optimization algorithm research, it remains a very challenging problem as how to effectively solve the large-scale optimization problem with global convergence capability. Thus with larger P, the optimization effectiveness may not be guaranteed, and the optimization results may be some local optimum which greatly affects the surrogate accuracy. With smaller P the optimization algorithm is more robust to obtain the global optimum. But it may fail to capture the high order polynomials of the true model. How to properly define P is a difficult problem. In this paper, it is proposed a pre-training method which conduct the multitask optimization under different highest order settings, i.e. $P = 1, 2, ..., P_{max}$, where P_{max} is a user-defined maximum, and obtain the preliminary objective values $\overset{\frown}{\boldsymbol{\Theta}}_k^P$, $k = 1, ..., K$ for each subset optimization task under different P values. Then for the kth subset, define the proper highest order as

$$P_k = \arg\min_P \{\overset{\frown}{\boldsymbol{\Theta}}_k^P\} \tag{14}$$

With P_k, the kth sub-model is trained, and the corresponding optimal active basis function vector $\mathbf{S}^{(-k)^*}$ could be obtained. The necessity of the highest order setting is illustrated in Sect. 4.3.1.

Ensemble modeling

How to combine the sub-models into the final ensemble with good performance is an important and challenging issue. Previous studies generally use the weighted sum approach, but it may lead to incorrect selection of basis functions due to the similar reasons for the PRS modeling based on total error sum of all the subsets, as analyzed in Sect. 2.3. Thus, the interpretability and accuracy of the ensemble are both reduced. To address this problem, instead of directly summing all the sub-models into an ensemble, it is proposed to construct the final model by quantitatively scoring and rationally selecting the basis functions in this paper.

First, a novel scoring method is proposed to measure the significance of each basis function. There are two important issues that should be taken into consideration during scoring, namely the sub-model accuracy and basis function occurrence frequency. On one hand, if a sub-model has high accuracy as well as low complexity (fewer active basis function numbers), then it would more possibly match the true model. Thus the active basis functions of this sub-model have more significance for surrogate modeling and should have a higher probability to be selected. On the other hand, if a basis function becomes active in many sub-models, it is more likely to be included in the true model for its high occurrence frequency among the sub-models. According to these considerations, the significance metric to score the basis function is defined as

$$\begin{aligned}
&\boldsymbol{\Phi}_s = \bigcup_{1 \le k \le K} \mathbf{S}^{(-k)^*} = \{f_i\}_{i=1,...,N_s} \\
&for \quad i = 1, 2, ..., N_s; \ k = 1, 2, ..., K \\
&score_{ik} = \begin{cases} \frac{1}{\overset{\frown}{\boldsymbol{\Theta}}_k^* \times N_{active}^{(-k)^*}} & \text{if} \quad f_i \in \mathbf{S}^{(-k)^*} \\ 0 & \text{if} \quad f_i \notin \mathbf{S}^{(-k)^*} \end{cases} \\
&score_i = \sum_{k=1}^{K} score_{ik}
\end{aligned} \tag{15}$$

where $\overset{\frown}{\boldsymbol{\Theta}}_k^*$ is the optimal objective value for the kth sub-model. $N_{active}^{(-k)^*}$ is the number of active basis functions in the kth sub-model. f_i denotes the ith component in the basis function set $\boldsymbol{\Phi}_s$ which is the union of all the candidate basis function sets $\mathbf{S}^{(-k)^*}$ of the sub-models, and N_s is the total number of basis functions in $\boldsymbol{\Phi}_s$. Each basis function $f_i (i = 1, 2, ..., N_s)$ is first scored in each sub-model and denoted as $score_{ik}$. Then the final score $score_i$ of each basis function is obtained by summing up $score_{ik}$ across all the sub-models. From Eq.(15), it can be observed that the following rules are applied for basis function selection:

- First, if the ith basis function is inactive in the kth sub-model, then the sub-score $score_{ik}$ of this basis function in this sub-model is zero. If it is active, then its sub-score is calculated according to the accuracy and complexity of this sub-model.
- Second, if a sub-model has smaller $\overset{\frown}{\boldsymbol{\Theta}}{}_k^*$, which means this sub-model's performance is good with high accuracy, then the active basis functions in this sub-model tend to have higher scores as this sub-model has larger possibility to match the true model.
- Third, if a sub-model has larger $N_{active}^{(-k)*}$, which means this sub-model is more complex with a large number of basis functions, then the active basis functions in this sub-model tend to have lower scores so as to prevent over-fitting.
- Fourth, through adding the sub-scores of basis functions across all the sub-models, the overall significant assessment of each basis function can be obtained with consideration for all the subsets.

After the scoring procedure, the sequence of the basis functions according to the significance scores in descending order can be obtained, and the top $N_0 = \min\,(N_s, n-2)$ elements are selected and composed the candidate significant basis function set $\{f_{(j)}\}_{j=1,\ldots,N_0}$. In this paper, it is proposed to add one basis function of $\{f_{(j)}\}_{j=1,\ldots,N_0}$ into the ensemble each time according to the ranking sequence and quantify the accuracy of this ensemble. After the jth basis function $f_{(j)}$ is added, there are j active basis functions in the current ensemble, based on which the PRS model can be built based on Sect. 2.1. Denote the prediction model of this ensemble as $\hat{y}_{(j)}(\boldsymbol{x})$. Then the bias-corrected Akaike's information criterion (AICc) (Hurvich and Tsai [35]) is used to quantify the model accuracy which is

$$\begin{aligned} \mathrm{AICc}_{(j)} &= n \times \ln\Big(\frac{1}{n}\sum_{l=1}^{n}\big(y^{(l)} - \hat{y}_{(j)}(\boldsymbol{x}^{(l)})\big)^2\Big) \\ &\quad + n \times (\ln(2\pi)+1) + 2j + 2j \times \frac{j+1}{n-j-1} \end{aligned} \tag{16}$$

Until all the basis functions are added into the ensemble, there are totally N_0 candidate ensemble models. By comparing $\mathrm{AICc}_{(j)}$ $(1 \le j \le N_0)$, the ensemble with the minimum AICc can be selected as the optimum model with the highest accuracy.

Evolutionary algorithm with transfer migration

In the PRS-MOEM framework, the efficacy of solving multitask optimization is very important. To enhance the optimization effectiveness, the evolutionary genetic algorithm (GA) is applied and improved in the following two aspects. First, the chromosomes of GA are coded to indicate the active states of basis functions. Second, a transfer migration method is proposed to exchange useful information between the different optimization tasks. The diagram of chromosome coding is shown in Fig. 3. For the optimization tasks, there are $nvars = (p+m)!/p!m!$ basis functions in the candidate set $\boldsymbol{\Phi}$. Then, compose the chromosome by $nvars$ bits with each bit corresponding to a basis function state. m is the dimension number of the input variable vector $\boldsymbol{x}$. $P = 1, 2, \ldots, P_{max}$ is the highest order. The bit with the value of 0 denotes that the corresponding basis function is inactive, while the bit with the value of 1 is the opposite. Note that searching the optimal chromosome is essentially to find the most proper active basis functions vector in the basis function set $\boldsymbol{\Phi}$.

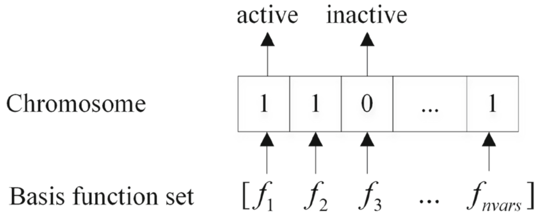


Fig. 3 The diagram of chromosome coding.

To improve the optimization performance, a one-way circular transfer migration strategy is proposed to bridge the parallel search with positive information exchange. Define population of the kth optimization task in the ith generation as $\text{population}_k^{(i)}$, which consists of three subpopulations $A_k^{(i)}, B_k^{(i)}, C_k^{(i)}$. Rank the population according to the objective value in ascending order for minimization problem (or descending order for maximization problem). $A_k^{(i)}$ is first $W\%$ elite of ranked population, and $C_k^{(i)}$ is last $W\%$ of ranked population. $B_k^{(i)}$ is the rest of the $\text{population}_k^{(i)}$. For $k = 1, 2, \ldots, K$, the $\text{population}_k^{(i)}$ is denoted as

$$\begin{aligned} &\text{population}_1^{(i)} = \{A_1^{(i)}, B_1^{(i)}, C_1^{(i)}\} \\ &\ldots \\ &\text{population}_k^{(i)} = \{A_k^{(i)}, B_k^{(i)}, C_k^{(i)}\} \\ &\ldots \\ &\text{population}_K^{(i)} = \{A_K^{(i)}, B_K^{(i)}, C_K^{(i)}\}. \end{aligned} \tag{17}$$

Based on the idea of positive transfer migration to enhance optimization efficiency, the multiple optimization tasks can be mutually boosted and accelerated by transferring the elite from task to task. The subpopulations between different optimization tasks migrate every G generation. For simplicity, the transfer migration of subpopulations only occurs in the adjacent optimization tasks. The diagram of the transfer migration between populations in different optimization tasks is shown in Fig. 5. For the optimization task $k = 1, 2, \ldots, K-1$, the top $W\%$ elite subpopulation $A_k^{(i)}$ are transferred to the task $k+1$ and used to replace the bottom

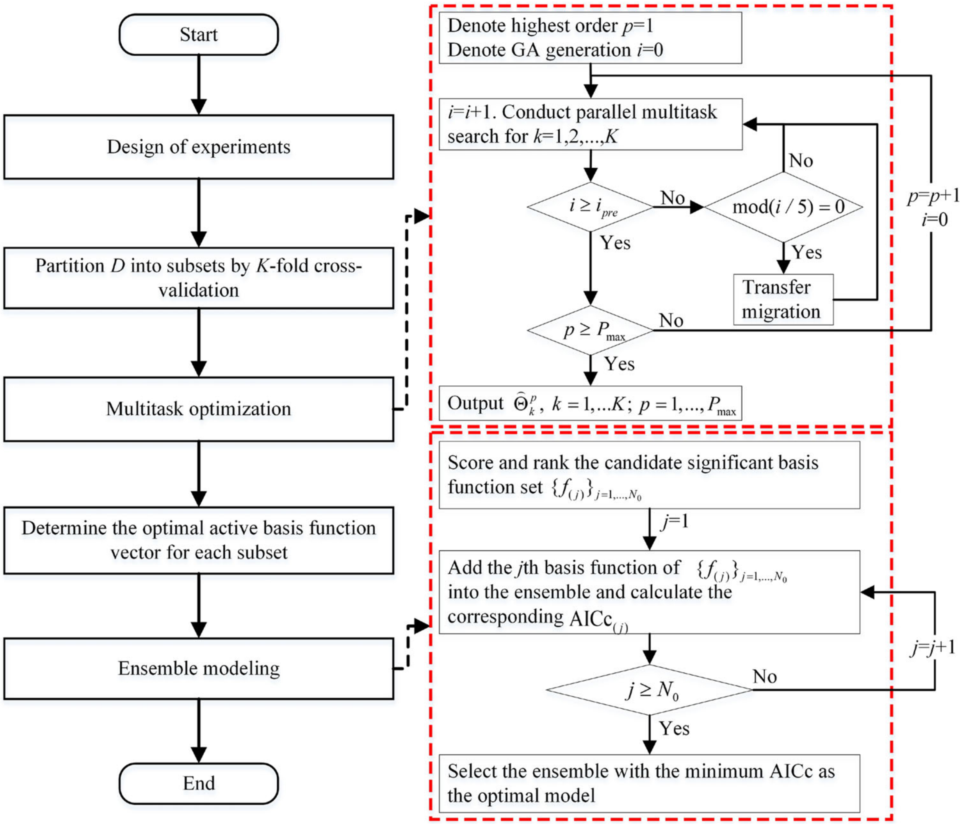

Start
Design of experiments
Partition D into subsets by K-fold cross-validation
Multitask optimization
Determine the optimal active basis function vector for each subset
Ensemble modeling
End
Denote highest order p=1
Denote GA generation i=0
i=i+1. Conduct parallel multitask search for k=1,2,...,K
$i \ge i_{pre}$
No
Yes
mod(i / 5) = 0
No
Yes
Transfer migration
$p \ge P_{\max}$
No
Yes
p=p+1
i=0
Output $\hat{\Theta}_k^p$, k = 1,...K; p = 1,..., $P_{\max}$
Score and rank the candidate significant basis function set $\{f_{(j)}\}_{j=1,...,N_0}$
j=1
Add the jth basis function of $\{f_{(j)}\}_{j=1,...,N_0}$ into the ensemble and calculate the corresponding AICc$_{(j)}$
$j \ge N_0$
No
Yes
j=j+1
Select the ensemble with the minimum AICc as the optimal model

Fig. 4 The flowchart of PRS-MOEM

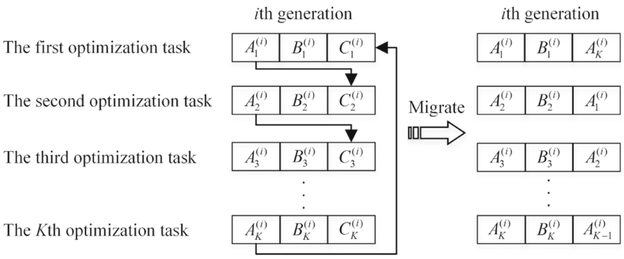

ith generation
ith generation
The first optimization task
The second optimization task
The third optimization task
The Kth optimization task
$A_1^{(i)}$ $B_1^{(i)}$ $C_1^{(i)}$
$A_2^{(i)}$ $B_2^{(i)}$ $C_2^{(i)}$
$A_3^{(i)}$ $B_3^{(i)}$ $C_3^{(i)}$
$A_K^{(i)}$ $B_K^{(i)}$ $C_K^{(i)}$
Migrate
$A_1^{(i)}$ $B_1^{(i)}$ $A_K^{(i)}$
$A_2^{(i)}$ $B_2^{(i)}$ $A_1^{(i)}$
$A_3^{(i)}$ $B_3^{(i)}$ $A_2^{(i)}$
$A_K^{(i)}$ $B_K^{(i)}$ $A_{K-1}^{(i)}$

Fig. 5 The diagram of the cyclical one-way transfer migration method

ranked subpopulation $C_{k+1}^{(i)}$. The elite subpopulation $A_K^{(i)}$ is transferred back to the task 1 and used to replace the bottom ranked subpopulation $C_1^{(i)}$. The transfer migration fraction $W\%$ is often set as 20%. The generation interval G for transfer migration operation is set as 5 in this paper. The GA search stops when the number of generations achieves a user-defined maximal value. To demonstrate the advantage of the proposed transfer migration, the performance with and without the transfer migration would be compared in Sect. 4.3.2.

Algorithm of PRS-MOEM

To sum up, based on the preceding multitask optimization and ensemble modeling framework, the flowchart of PRS-MOEM is shown in Fig. 4, and the detailed steps are explained as follows:

– Step 1: Design of experiments (DOE). Obtain the training point set $D = \{(\mathbf{x}^{(l)}, y^{(l)}), l = 1, 2, ..., n\}$ with DOE methods, e.g. Latin hypercube design (LHD) (Dette and Pepelyshev [36]).
– Step 2: Partition D into subsets by K-fold cross-validation. The training set $D = \{(\mathbf{x}^{(l)}, y^{(l)}), l = 1, 2, ..., n\}$ is randomly partitioned into K disjoint sets of approximately equal size, denoted as $D_1, D_2, ..., D_K$. For $k = 1, 2, ..., K$, denote the training subset as $D^{(-k)} = D - D_k$ and the validation subset as D_k.
– Step 3: Multitask optimization. For different highest order settings $P = 1, 2, ..., P_{max}$, compose the candidate basis function set and conduct the multitask optimization. GA is used as the optimization solver and the top $W\%$ elite subpopulation migrate between different optimization tasks every G generations. In this paper, $W = 20$ and $G = 5$. GA stops when the number of generations i achieves a user-defined preliminary value i_{pre}. In this paper, to save the computational time and define the proper highest order, $i_{pre} = 10$.
– Step 4: Determine the optimal active basis function vector for each subset. For the kth optimization ($k = 1, 2, ..., K$), define the proper highest order P_k by Eq.(14)

which has the minimum objective values $\overset{\frown}{\boldsymbol{\Theta}}_k^P$. Then with P_k, the kth sub-model is trained using multitask optimization in Step 3 from the number of generations i_{pre} to a user-defined maximal values $i_{\max}$, and accordingly obtain the optimal active basis function vector $\mathbf{S}^{(-k)^*}$.

- Step 5: Ensemble modeling. Based on the optimal active basis function vectors $\mathbf{S}^{(-k)^*}$, compose the candidate active basis function set $\boldsymbol{\Phi}_s = \bigcup\limits_{1\le k\le K} \mathbf{S}^{(-k)^*}$ and denote the total elements number as N_s. Score the significance of each basis function of this set by Eq.(15), and rank the set in the descending order according to the scores. Select the top $N_0 = \min(N_s, n-2)$ elements to compose the candidate significant basis function set $\{f_{(j)}\}_{j=1,...,N_0}$. Each time add one basis function of $\{f_{(j)}\}_{j=1,...,N_0}$ into the ensemble according to the ranking sequence and quantify the accuracy of this ensemble with AICc. For $j = 1, ..., N_0$, add the basis function $f_{(j)}$ into the ensemble and calculate the accuracy $\text{AICc}_{(j)}$ of the corresponding PRS model. Select the ensemble with the minimum AICc as the optimal model with the highest accuracy.

Numerical examples

Test problem

To illustrate the effectiveness of the proposed method in this paper, two types of problems are chosen: a). benchmark numerical test functions (Jamil and Yang [37]) and b). a practical engineering problem for sandwich panel design. The details are given below:

1. Branin function:

$$
\begin{aligned}
&f(\boldsymbol{x}) = (x_2 - \tfrac{5.1x_1^2}{4\pi^2} + \tfrac{5x_1}{\pi} - 6)^2 \\
&\qquad +10(1 - \tfrac{1}{8\pi})\cos(x_1) + 10 \\
&x_1 \in [-5, 10], x_2 \in [0, 15]
\end{aligned}
\tag{18}
$$

2. Three-Hump function:

$$
\begin{aligned}
&f(\boldsymbol{x}) = 2x_1^2 - 1.05x_1^4 + \tfrac{x_1^6}{6} + x_1x_2 + x_2^2 \\
&x_1, x_2 \in [-2, 2]
\end{aligned}
\tag{19}
$$

3. Giunta function:

$$
\begin{aligned}
&f(\boldsymbol{x}) = 0.6 + \textstyle\sum_{i=1}^2 \begin{bmatrix} \sin(\frac{16x_i}{15}-1) + \sin^2(\frac{16x_i}{15}-1) \\ + \sin(4(\frac{16x_i}{15}-1))/50 \end{bmatrix} \\
&x_1, x_2 \in [-1, 1]
\end{aligned}
\tag{20}
$$

4. Schaffer function:

$$
\begin{aligned}
&f(\boldsymbol{x}) = 0.5 + \tfrac{(\sin((x_1^2+x_2^2)^{\frac{1}{2}}))^2-0.5}{1+0.001\times(x_1^2+x_2^2)} \\
&x_1 \in [-3, 3], x_2 \in [-3, 3]
\end{aligned}
\tag{21}
$$

5. Biggs function:

$$
\begin{aligned}
&f(\boldsymbol{x}) = \sum_{i=1}^{10} (e^{-t_i x_1} - x_3 e^{-t_i x_2} - y_i)^2 \\
&t_i = 0.1i, y_i = e^{-t_i} - 5e^{-10t_i}, i = 1, ..., 10 \\
&x_j \in [0, 20], j = 1, 2, 3
\end{aligned}
\tag{22}
$$

6. Dette and Pepelyshev curved (DP3) function:

$$
\begin{aligned}
&f(\boldsymbol{x}) = 4(x_1 - 2 + 8x_2 - 8x_2^2)^2 + (3 - 4x_2)^2 \\
&\qquad +16\sqrt{x_3+1}(2x_3 - 1)^2 \\
&x_i \in [0, 1], i = 1, 2, 3
\end{aligned}
\tag{23}
$$

7. Colville function:

$$
\begin{aligned}
&f(\boldsymbol{x}) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2 + (x_3 - 1)^2 \\
&+90(x_3^2 - x_4)^2 + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2] \\
&+19.8(x_2 - 1)(x_4 - 1) \\
&x_i \in [-10, 10], i = 1, 2, 3, 4
\end{aligned}
\tag{24}
$$

The structural design of all-metal sandwich panels has great success in many engineering applications (Stickel and Nagarajan [38]), and the global deflection of the sandwich panels is an essential structural response in the optimal design. However, with finite element methods and natural tests, the calculation of the global deflection is time-consuming and quite complex. To save the computation cost, surrogate modeling methods are often applied which could provide simple but reliable metamodels. In this paper, a square-core type sandwich panel design problem is selected to investigate the performance of the proposed method in

Table 1 The description of the design variables for the sandwich panel design problem

Variable	Description	Range	Units
L	Panel length	[3,7]	m
h	Panel height	[4,16]	mm
t_1	Top and bottom plate thickness	[2,4]	mm
t_2	Core stiffener thickness	[1.5,4]	mm
k_h	Core stiffener spacing factor	[1.5,4]	
n	Symmetrical number of core stiffeners	[2,6]	

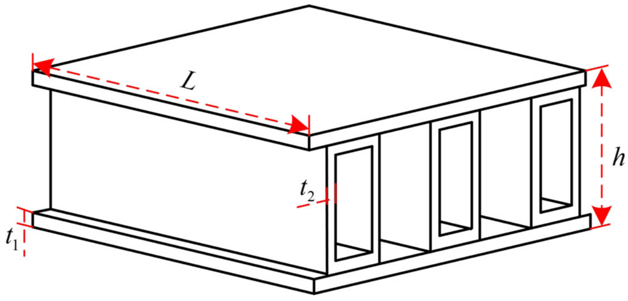


Fig. 6 The square-core type sandwich panel

the field of engineering (Kalnins et al. [39]), which is shown in Fig. 6. The global deflection of the panel is determined by six design variables, the description of which are outlined in Table 1. The response values of global deflection are calculated by FEM commercial software ANSYS employing SHELL 181-4-node shell element, more details can be found in the literature (Kalnins et al. [39]). The simulation data are downloaded from http://www.cs.rtu.lv/jekabsons/datasets.html and used in this paper for surrogate modeling.

Experimental settings

To verify the proposed PRS-MOEM method, it is compared with the following popular surrogates: 1). Polynomial Response Surface (PRS), 2). Least absolute shrinkage and selection (Lasso), 3). Ordinary Kriging (OK), 4). Radial Basis Function (RBF), 5). Elastic Net (EN) (Li et al. [41]), and 6). The traditional PRS modeling method based on cross validation (PRS-CCV) as introduced in Sect. 2.2. With the reference to (Zhang et al. [10]; Yin et al. [11]; Fan et al. [24]; Gribonval et al. [40]; Li et al. [41]), the parameter settings are presented in Table 2.

Root mean squared error (RMSE) and Maximum absolute error (MAE) (Goel et al. [18]) are used to evaluate the predictive capabilities of the surrogate models in this paper, which are

$$\mathrm{RMSE} = \sqrt{\frac{1}{n_t}\sum_{l=1}^{n_t}\left(y^{(l)} - \hat{y}(\boldsymbol{x}^{(l)})\right)^2} \tag{25}$$

$$\mathrm{MAE} = \max|y^{(l)} - \hat{y}(\boldsymbol{x}^{(l)})|, l = 1, 2, ..., n_t \tag{26}$$

where $y^{(l)}$ and $\hat{y}(\boldsymbol{x}^{(l)})$ denotes the actual response and the predicted response at the lth test point, respectively, and n_t is the number of test points. For high-quality surrogate model, RMSE and MAE should be both low.

For all the benchmark numerical examples in Eq.(18)–(24), Latin hypercube design (LHD) is used for sampling the training data set. For the sandwich panel design problem, the sample set of 500 points analyzed by finite element simulation are downloaded from the online web source. The data set is divided into 60 training points and 440 test points randomly. Considering the effect of random sampling, 100 training sets are obtained for each test problem randomly, as shown in Table 3, based on which the surrogate modeling is conducted repeatedly and independently. Then the statistical performance (the mean and variation of RMSE and MAE) of different surrogate models can be investigated. The variation of each prediction metric is depicted with box plots. A short tail and a small size of the box plot signify a robust approximation. In addition, the computational time (CPU time) of the different surrogate modeling methods are also given, the tests are performed on a personal computer with a 2.3GHz CPU and 8GB RAM.

Result and discussion

Determination of the highest order

To illustrate the necessity of determining the highest order, the optimization results of multitask optimization under different highest order settings $P = 1, 2, ..., 10$ for Schaffer and Biggs functions are shown in Tables 4 and 5 respectively. The lowest value in each column is shown in bold for ease of comparison. $\overset{\frown}{\boldsymbol{\Theta}}{}_k^*, k = 1, ..., 10$ denotes the optimal objective value for the kth sub-model. For the Schaffer test function, it is observed that $P = 8$ is the best for the first, fourth, sixth, seventh, eighth, and tenth subsets, and $P = 9$ is the best for the second, third, fifth, and ninth subsets. Similarly, the most proper highest order for Biggs test function is $P = 5$ or 6 for different subsets. The results show that the optimum obtained by GA is greatly influenced by the highest

Table 2 Parameters for different surrogate models

Surrogate model	Details
PRS	The highest order of polynomials is $P = 2$.
Lasso	The number of basis functions is set to $6n$ (where n is the number of sample points) (Gribonval et al. [40]). The 10-fold cross-validation method is used to choose the regularization parameter λ.
OK	The constant regression function and Gaussian correlation model are employed. In all cases, $\theta_0 = 1_{m\times 1}$, and $0.1 \le \theta_i \le 20$ (Gribonval et al. [10]), for $i = 1, 2, ..., m$, where m is the number of variables and $1_{m\times 1}$ is the vector whose entries are all equal to 1.
RBF	The basis function is multiquadric with $c = 0.9$.
EN	The number of basis functions is set as $6n$. The ridge regularization parameter is $\lambda_2 = 10e - 3$ (Li et al. [41]), and the 10-fold cross-validation method is used to determine the lasso regularization parameter.
PRS-MOEM	10-fold cross-validation is used to partition the training set, and P_{max} is set to 10.
PRS-CCV	The settings are the same as PRS-MOEM.

Table 3 Numerical setup for test problems

Test problem	No. of variables	No. of training points	No. of test points	No. of training sets
Branin	2	30	1000	100
Three-Hump	2	30	1000	100
Giunta	2	30	1000	100
Schaffer	2	30	1000	100
Biggs	3	40	1331	100
DP3	3	40	1331	100
Colville	4	50	2400	100
Sandwich panel	6	60	440	100

Table 4 Optimal objective values of different highest order settings in Schaffer test

	$\overset{\frown}{\boldsymbol{\Theta}}{}_1^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_2^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_3^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_4^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_5^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_6^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_7^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_8^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_9^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_{10}^*$
p=1	0.3535	0.2958	0.2580	0.2608	0.3215	0.3313	0.3417	0.3316	0.3214	0.3129
p=2	0.2196	0.1779	0.1608	0.1781	0.1995	0.1941	0.2418	0.2256	0.2118	0.2149
p=3	0.1718	0.1233	0.1032	0.1227	0.1511	0.1281	0.1991	0.1711	0.1546	0.1585
p=4	0.1385	0.0933	0.0783	0.0987	0.1251	0.1000	0.1710	0.1459	0.1286	0.1364
p=5	0.0949	0.0494	0.0422	0.0430	0.0684	0.0472	0.1134	0.0874	0.0753	0.0835
p=6	0.0104	0.0107	0.0131	0.0141	0.0164	0.0108	0.0156	0.0117	0.0122	0.0141
p=7	0.0121	0.0094	0.0090	0.0129	0.0087	0.0113	0.0107	0.0108	0.0122	0.0104
p=8	**0.0083**	0.0087	0.0098	**0.0104**	0.0080	**0.0081**	**0.0096**	**0.0079**	0.0090	**0.0085**
p=9	0.0086	**0.0070**	**0.0079**	0.0107	**0.0070**	0.0085	0.0120	0.0105	**0.0082**	0.0109
p=10	0.0099	0.0101	0.0093	0.0139	0.0105	0.0105	0.0112	0.0093	0.0079	0.0097

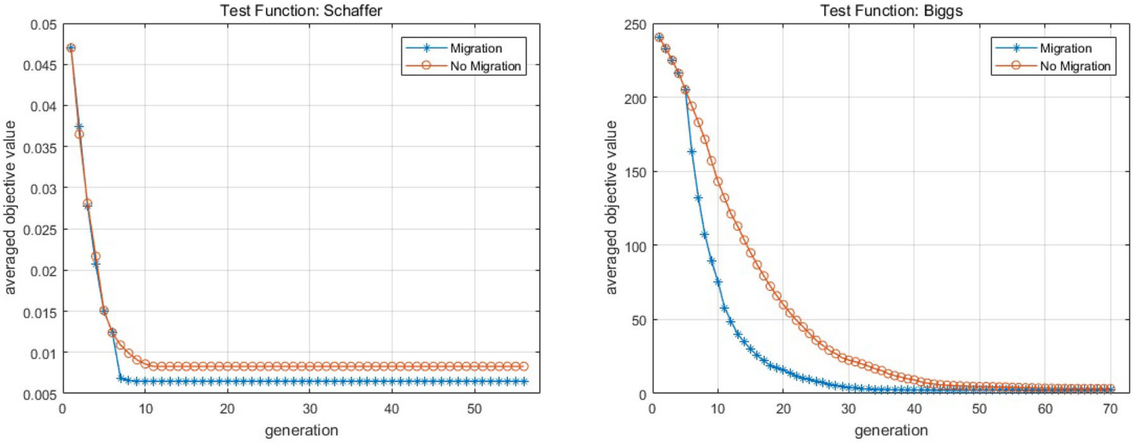
order settings. Take the Biggs test for example. When P is increased from one to five, the optimal objective values of all subsets are dramatically reduced by two orders. It clearly indicated that with small P values, the high order nonlinearity of the test function cannot be captured by the PRS model, which leads to a large prediction error. When $P = 5$ or 6, the optimal objective values of different subsets reach their corresponding lowest point. The variation of the best P values among different optimization tasks mainly results from two issues. One is the randomness of subset partition, which leads to different features observed from different training and validation subsets. The other is the inherent fluctuation associated with GA. For the same subset, the differences between the optimal objective values for $P = 5$ and 6 are

Table 5 Optimal objective values of different highest order settings in Biggs test

	$\overset{\frown}{\boldsymbol{\Theta}}{}_1^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_2^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_3^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_4^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_5^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_6^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_7^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_8^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_9^*$	$\overset{\frown}{\boldsymbol{\Theta}}{}_{10}^*$
p=1	80.480	29.719	44.188	37.597	69.097	31.865	56.890	43.688	43.987	52.452
p=2	50.496	16.863	21.530	23.027	38.039	14.256	32.230	22.262	23.478	31.537
p=3	23.434	8.099	9.235	8.062	14.977	4.824	15.798	9.091	7.216	13.712
p=4	3.003	1.765	1.508	1.502	1.357	2.195	2.175	1.430	1.777	1.431
p=5	**0.677**	**1.060**	**0.941**	**0.802**	0.933	**0.677**	0.857	**0.684**	**0.789**	**0.890**
p=6	0.859	1.252	1.100	0.897	**0.887**	0.834	**0.855**	0.802	0.874	0.896
p=7	1.442	1.551	1.558	1.664	1.633	1.424	1.496	1.438	1.715	1.902
p=8	390.260	419.350	401.280	341.440	449.220	388.470	487.460	356.290	448.270	398.710
p=9	362.250	404.040	419.130	356.190	440.090	370.120	386.590	417.470	325.820	371.260
p=10	350.290	330.060	330.020	311.580	333.840	322.640	376.070	268.100	327.370	361.600

Table 6 Comparison of the average optimal objective values for Schaffer and Biggs tests

Function		$E(\overset{\frown}{\boldsymbol{\Theta}}{}_1^*)$	$E(\overset{\frown}{\boldsymbol{\Theta}}{}_2^*)$	$E(\overset{\frown}{\boldsymbol{\Theta}}{}_3^*)$	$E(\overset{\frown}{\boldsymbol{\Theta}}{}_4^*)$	$E(\overset{\frown}{\boldsymbol{\Theta}}{}_5^*)$	$E(\overset{\frown}{\boldsymbol{\Theta}}{}_6^*)$	$E(\overset{\frown}{\boldsymbol{\Theta}}{}_7^*)$	$E(\overset{\frown}{\boldsymbol{\Theta}}{}_8^*)$	$E(\overset{\frown}{\boldsymbol{\Theta}}{}_9^*)$	$E(\overset{\frown}{\boldsymbol{\Theta}}{}_{10}^*)$
Schaffer	Migration	0.0065	0.0069	0.0068	0.0074	0.0075	0.0071	0.0074	0.0084	0.0069	0.0069
	No Migration	0.0083	0.0087	0.0079	0.0077	0.0077	0.0085	0.0075	0.0088	0.0073	0.008
Biggs	Migration	2.243	2.038	2.327	1.896	2.023	1.824	1.884	2.821	2.493	3.144
	No Migration	3.135	2.911	2.489	2.187	3.034	2.608	2.436	3.375	3.106	3.195



(a) Schaffer

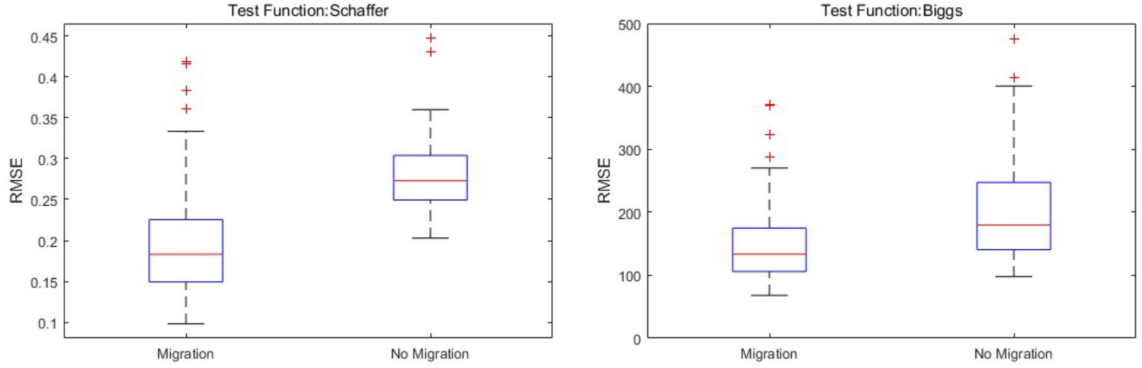
(b) Biggs

Fig. 7 The average convergence process for Schaffer and Biggs tests

very small, which indicates the stableness of GA in solving the training optimization problems for the Biggs function at this dimension. Then with the continued increase of P, the optimal objective values dramatically increase again due to the failure of GA to obtain the global optimum with the fast increase of the design space dimension. Thus the most proper highest order for the Biggs test is determined as $P = 5$ or 6 for different subsets, according to which the best active basis function vector can be selected.

Effect of the transfer migration method

To demonstrate the advantage of the transfer migration, multitask optimization is conducted with and without the step of the transfer migration for the Schaffer test with the highest order setting $P = 8$ and for the Biggs test functions with $P = 5$ respectively. The average convergence graphs of the first optimization task for subset one obtained by 100 independent runs are shown in Fig. 7. Besides, the average optimal objective values for ten optimization tasks obtained by the 100 independent runs are shown in Table 6.



(a) Schaffer

(b) Biggs

Fig. 8 The comparison between the RMSE boxplots obtained with and without transfer migration for Schaffer and Biggs tests

Table 7 Comparison between the average RMSE obtained with and without transfer migration

	Schaffer	Biggs
Migration	0.1986	149.392
No Migration	0.2785	286.487

Table 8 Comparison of the average RMSE in Branin and Schaffer tests

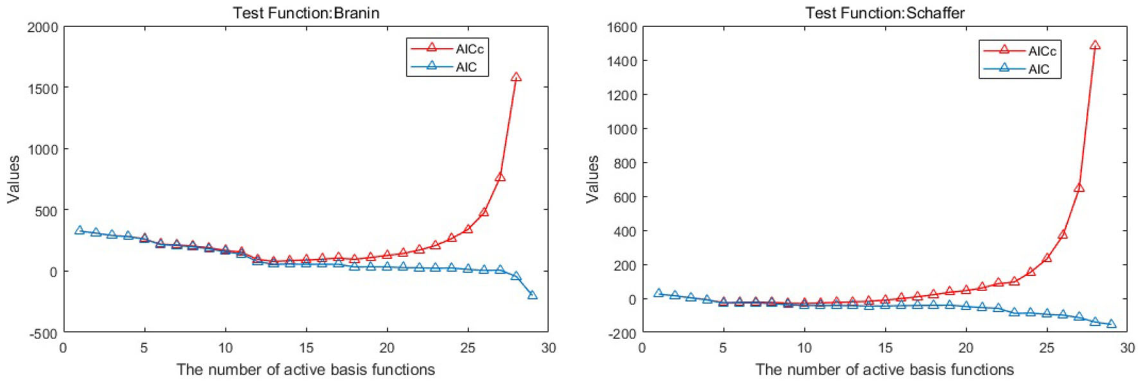
	Branin	Schaffer
The model using AICc	6.3982	0.1804
The model using AIC	9.4983	0.4621

$E(\overset{\frown}{\Theta}{}_k^{*})$, $k = 1, \ldots, 10$ denotes the average optimal objective value for the kth optimization task. It can be observed that for both Schaffer and Biggs test, the GA with the transfer migration converges faster, and its average optimal objective values are universally better than that without the transfer migration. These confirm the proposed method can enhance the optimization efficiency by means of positive transfer learning. On the other hand, the results show that the randomness of the subset partition could influence the optimization results greatly. Take the Biggs test for example. The maximal optimum is 3.144 for the tenth optimization task, and the minimal optimum is 1.824 for the sixth optimization task. To further investigate the effect of the proposed method on the prediction accuracy of the model, PRS-MOEM is constructed with and without the transfer migration. Box plots of the RMSE and the mean RMSE results obtained by the 100 independent runs for these two test functions are shown in Fig. 8 and Table 7. It could be easily seen that the model with the transfer migration performs much better than that without the transfer migration for both two tests, which verifies the effectiveness of the proposed multitask optimization algorithm in enhancing PRS modeling.

The above experiments clearly show the significance of the transfer migration method. However, it is noteworthy that without priority information, it is still a challenging problem that whether the transfer between different optimization tasks is positive or deleterious in practical engineering. In this paper, the specific advantage is that the similarity of multiple optimization tasks is ensured due to the training sample subsets being obtained from the same true model. Thus, the prediction accuracy of the model can be improved through the useful information exchange between these related optimization tasks.

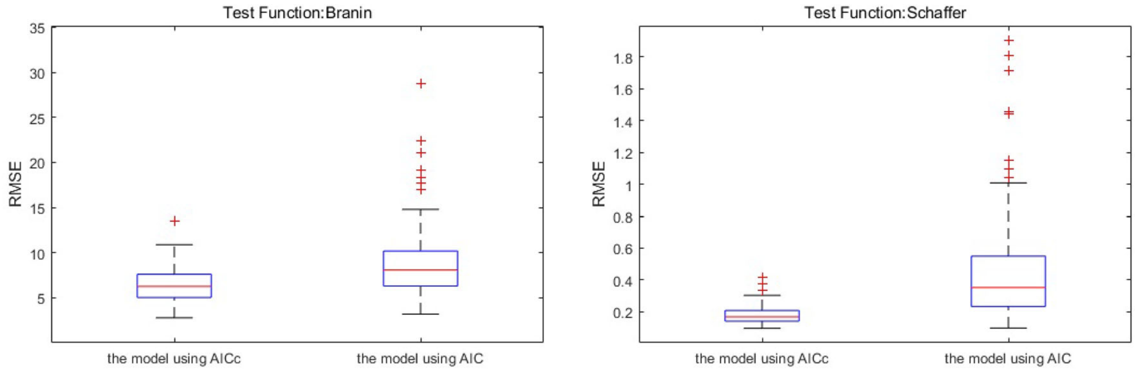
Effect of the model selection criterion

To investigate the rationality of using AICc instead of AIC to quantify the model accuracy in this paper, AICc and AIC are applied to select the final model for Branin and Schaffer test functions. Box plots of the RMSE and the mean RMSE results obtained by 100 independent runs are shown in Fig. 10 and Table 8 respectively. It can be observed that when AICc is chosen as the model selection criterion the box plot has a smaller size and shorter tail, and the mean RMSE of the models using AICc is much lower than that using AIC for both two test functions. It suggests that AICc can select the final model with better performance. To further study the effect of using AIC and AICc on basis function selection respectively, the curves of the measure values of the two criteria with respect to the number of selected active basis functions are shown in Fig. 9. It could be seen that the value of AIC decreases gradually as the number of active basis functions increases, but the value of AICc decreases first and then increases significantly with the obvious turning point. It suggests that AIC tends to choose more basis functions, which could lead to the problem of over-fitting and result in lower prediction accuracy. In the contrast, AICc can prevent this risk effectively.



(a) Branin (b) Schaffer

Fig. 9 Values of AICc and AIC varying with the number of active basis functions in Branin and Schaffer test



(a) Branin (b) Schaffer

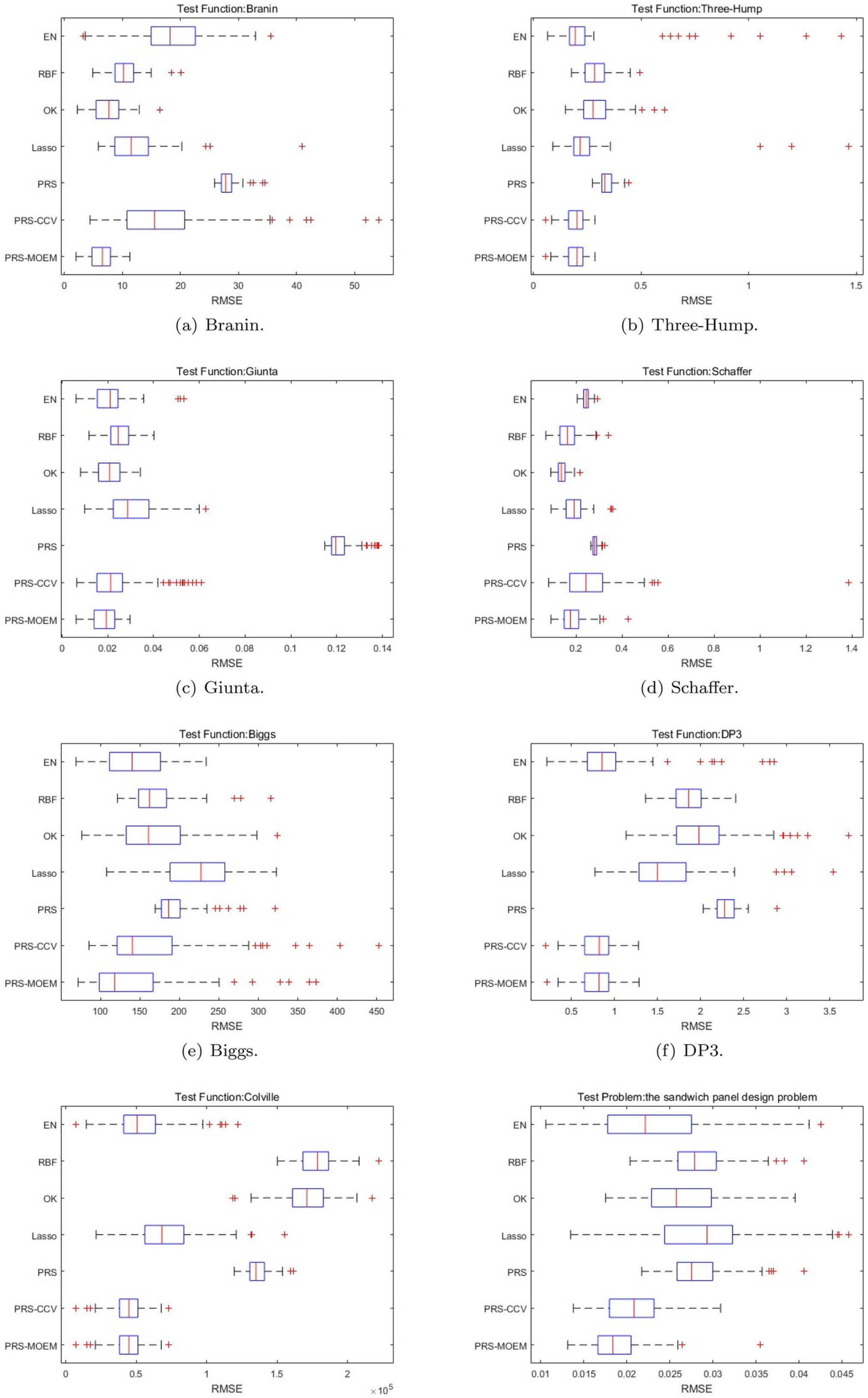
Fig. 10 Boxplots of RMSE for the model using AICc and the model using AIC in Branin and Schaffer tests

Accuracy and robustness

The accuracy of the different models is evaluated by the average prediction metrics of the 100 independent runs. Tables 9 and 10 show the mean RMSE and MAE for the different surrogate models respectively. The lowest value in each column is shown in bold for ease of comparison. Moreover, to further evaluate the robustness of different models, the variation of RMSE and MAE is expressed with box plots in Figs. 11 and 12 respectively. The RMSE and MAE results show that PRS-MOEM performs the best among the surrogates in all the test problems except for the Schaffer test function. For Three-Hump, DP3, and Colville problems, PRS-CCV also performs well and the accuracy results are very close to PRS-MOEM. However, for Branin, Giunta, Biggs, and the sandwich panel design problem, PRS-MOEM performs much better than PRS-CCV. To investigate the differences of the selected active basis functions between these two models, the histograms that describe the frequency of active basis functions obtained for the 100 training data sets of the Branin and Giunta functions are shown in Fig. 13 respectively. It can be seen that for PRS-CCV, except for some specific basis functions, the others are all selected with similar frequencies, which means PRS-CCV fails to identify the significant basis functions accurately and stably under the randomly generated training sets. In the contrast, for PRS-MOEM, there is a clear distinction of the frequencies between the significant and the other basis functions, which further verified the robustness of the proposed method.

Computational time

To show the computational cost of the different models, the seven surrogates are repeatedly constructed for the different test functions, Table 11 presents their average CPU time in seconds obtained by 100 independent runs. It can be seen that the average calculation time of PRS, OK, and RBF are universally small for all test functions, Lasso and EN have slightly larger calculation time. For PRS-MOEM and PRS-CCV, they spend more time building a model, around 2 seconds, which is much larger than the other surrogate modeling methods. But in practical engineering, this computational burden is negligible compared to the high-fidelity analysis of the training samples, e.g. it takes hundreds of



(a) Branin.

(b) Three-Hump.

(c) Giunta.

(d) Schaffer.

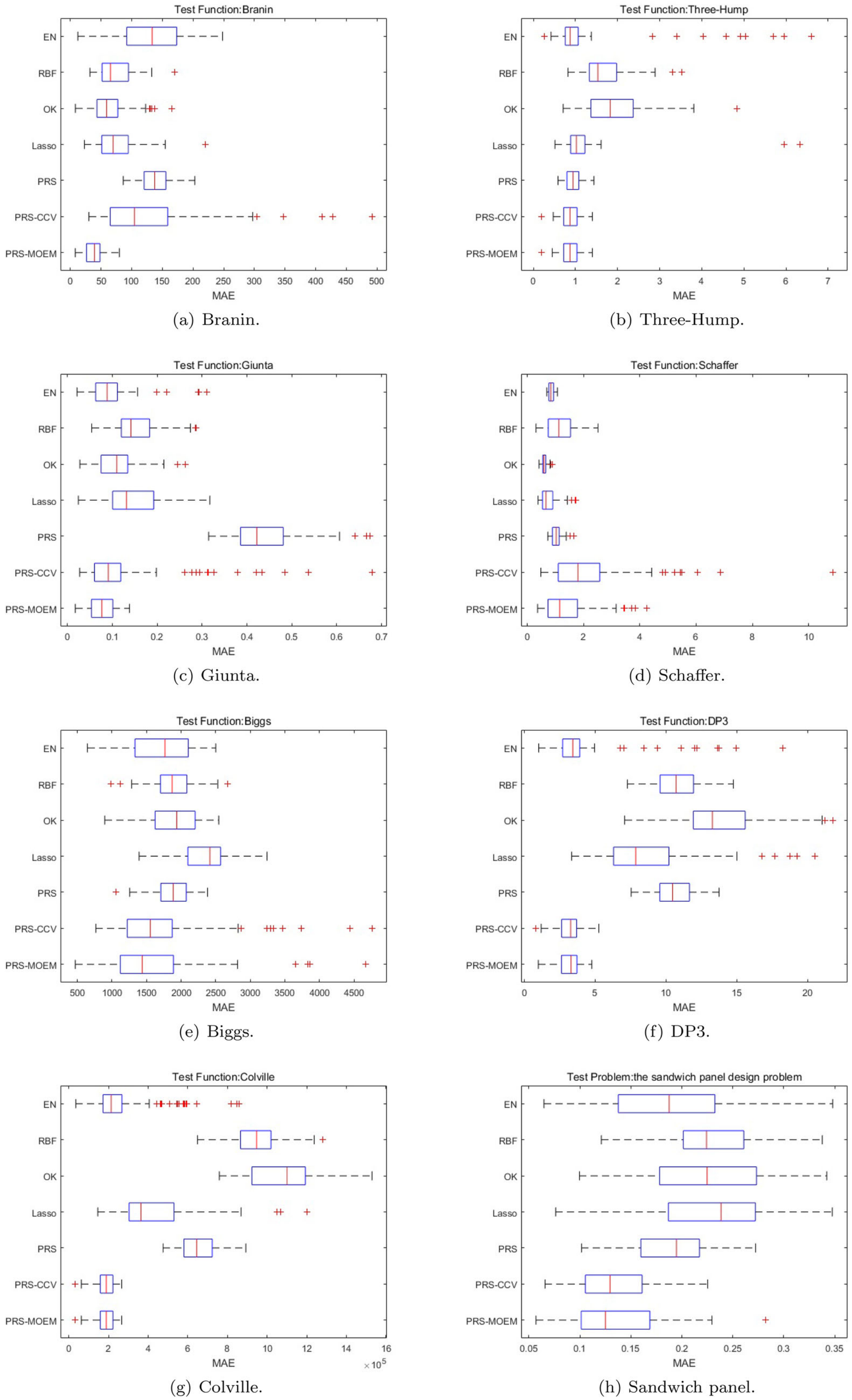
(e) Biggs.

(f) DP3.

(g) Colville.

(h) Sandwich panel.

Fig. 11 Box plots of RMSE for different surrogate models in **a** Branin, **b** Three-Hump, **c** Giunta, **d** Schaffer, **e** Biggs, **f** DP3, **g** Colville, and **h** the sandwich panel design problem



(a) Branin.

(b) Three-Hump.

(c) Giunta.

(d) Schaffer.

(e) Biggs.

(f) DP3.

(g) Colville.

(h) Sandwich panel.

Fig. 12 Box plots of MAE for different surrogate models in **a** Branin, **b** Three-Hump, **c** Giunta, **d** Schaffer, **e** Biggs, **f** DP3, **g** Colville, and **h** the sandwich panel design problem

Table 9 Comparison of the average RMSE for different surrogate models

	Branin	Three-Hump	Giunta	Schaffer	Biggs	DP3	Colville	SandwichPanel
PRS-MOEM	**6.3176**	**0.1896**	**0.0185**	0.1819	**140.85**	**0.804**	**4.39E+04**	**0.0188**
PRS-CCV	17.4893	0.1899	0.0244	0.2728	167.05	0.8068	4.39E+04	0.0208
PRS	28.1543	0.3322	0.1216	0.2819	193.47	2.2944	1.36E+05	0.0281
Lasso	12.1976	0.2333	0.0306	0.1953	219.03	1.6137	7.19E+04	0.029
OK	7.5634	0.2853	0.0205	**0.1377**	167.58	2.0293	1.71E+05	0.0264
RBF	10.3156	0.2845	0.0251	0.167	171.16	1.8659	1.78E+05	0.0282
EN	18.1909	0.2545	0.0209	0.2428	146.17	0.942	5.40E+04	0.0228

Table 10 Comparison of the average MAE for different surrogate models

	Branin	Three-Hump	Giunta	Schaffer	Biggs	DP3	Colville	SandwichPanel
PRS-MOEM	**37.328**	**0.896**	**0.077**	1.391	**1578.04**	**3.141**	**1.84E+05**	**0.1321**
PRS-CCV	129.692	0.897	0.123	2.169	1720.47	3.152	1.84E+05	0.1349
PRS	139.968	0.929	0.435	1.031	1880.29	10.567	6.57E+05	0.1874
Lasso	74.156	1.1	0.143	0.75	2336.34	8.615	4.38E+05	0.2315
OK	64.926	1.991	0.11	**0.601**	1915.62	13.703	1.08E+06	0.2232
RBF	72.273	1.691	0.153	1.175	1874.1	10.899	9.45E+05	0.2283
EN	130.229	1.275	0.093	0.85	1734.27	4.102	2.70E+05	0.186

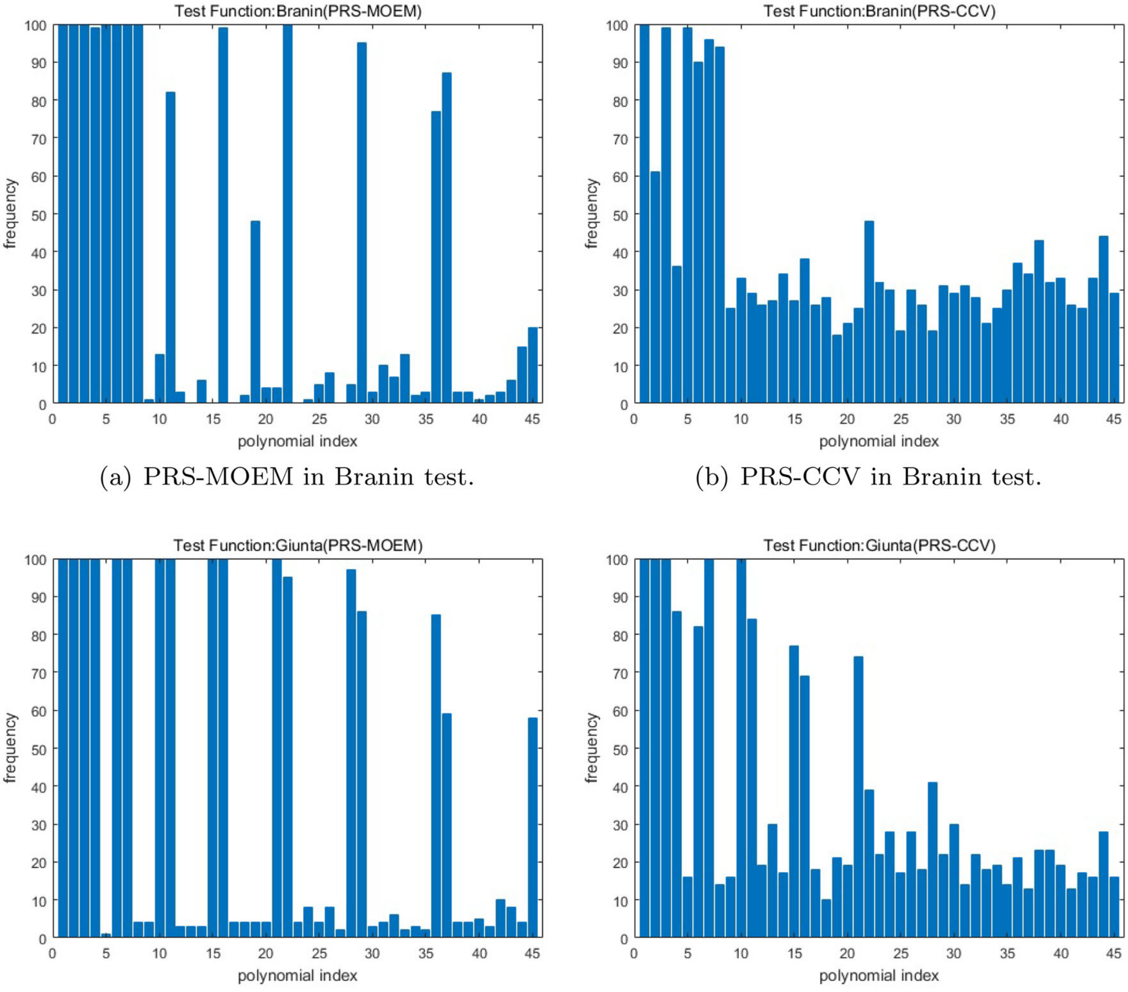
Table 11 Comparison of the average computational time (in seconds) for different surrogate models

	Branin	Three-Hump	Giunta	Schaffer	Biggs	DP3	Colville	SandwichPanel
PRS-MOEM	1.571	1.569	1.519	1.533	2.059	1.845	2.010	2.098
PRS-CCV	2.079	1.547	1.798	1.931	2.075	1.930	2.094	2.220
PRS	0.006	0.003	0.003	0.002	0.005	0.005	0.004	0.008
Lasso	0.067	0.064	0.061	0.062	0.106	0.145	0.110	0.564
OK	0.007	0.007	0.006	0.008	0.016	0.017	0.026	0.033
RBF	0.005	0.006	0.005	0.007	0.026	0.031	0.031	0.008
EN	0.062	0.057	0.052	0.056	0.153	0.155	0.180	0.150

hours to run a single finite element analysis for a complex flight vehicle structure. Thus for PRS modeling, it is worthwhile to spend modest more computational cost in selecting proper active basis function vector for better surrogate accuracy. In the future, more effective evolutionary algorithms should be studied to enhance the global optimization capability in solving large-scale optimization problems.

Conclusion

In this paper, an improved PRS modeling approach PRS-MOEM is proposed to enhance the prediction accuracy by better selecting basis functions with robustness based on the multitask optimization and ensemble modeling framework. Unlike the traditional PRS modeling based on cross validation which directly builds a single model guided by the total error estimation of all the training subsets, the proposed method constructs a sub-model for each training subset. By properly integrating all the sub-models, the information of the subsets can be fully explored. To construct all the sub-models effectively by solving the multitask optimization problem, an improved evolutionary algorithm with transfer migration is developed, which can significantly enhance the optimization efficiency and robustness by useful information exchange between similar optimization tasks. To obtain a well performed ensemble based on all the sub-models, a scoring method is proposed to measure the significance of each basis function according to the error estimation of the sub-models and the occurrence frequency of these basis functions in all the sub-models. Then based on the AICc criterion, the significant basis functions can be selected and the optimal ensemble can be defined. PRS-MOEM can effectively mitigate the negative influence from the sub-models with large prediction error, and alleviate the uncertain impact resulting from the



(a) PRS-MOEM in Branin test.

(b) PRS-CCV in Branin test.

(c) PRS-MOEM in Giunta test.

(d) PRS-CCV in Giunta test.

Fig. 13 Frequency of active basis functions for PRS-MOEM and PRS-CCV in Branin and Giunta tests

randomness of training subsets. Thus the basis function selection accuracy and robustness can be effectively enhanced, which are verified by seven numerical examples and one practical engineering application example in the test section. For future works, the applicability of PRS-MOEM will be further investigated on a more diverse set of test problems as well as complex practical engineering problems. Besides, the extension of PRS-MOEM to effectively solve high dimensional problems is also a promising research direction.

Acknowledgements This work was supported in part by National Natural Science Foundation of China under Grant No.11725211.

Compliance with ethical standards

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

References

1. Forrester AI, Keane AJ (2009) Recent advances in surrogate-based optimization. Prog Aerosp Sci 45(1–3):50–79. https://doi.org/10.1016/j.paerosci.2008.11.001
2. Namura N, Shimoyama K, Jeong S, Obayashi S, (2011) Kriging, RBF-hybrid response surface method for highly nonlinear func-

tions. (2011) IEEE Congress of Evolutionary Computation. CEC 2011(6):2534–2541. https://doi.org/10.1109/CEC.2011.5949933

3. Yao W, Chen X, Ouyang Q, Van Tooren M (2012) A surrogate based multistage-multilevel optimization procedure for multidisciplinary design optimization. Struct Multidiscip Optim 45(4):559–574. https://doi.org/10.1007/s00158-011-0714-z
4. Yao W, Chen X, Ouyang Q, Van Tooren M (2013) A reliability-based multidisciplinary design optimization procedure based on combined probability and evidence theory. Struct Multidiscip Optim 48(2):339–354. https://doi.org/10.1007/s00158-013-0901-1
5. Goel T, Hafkta RT, Shyy W (2009) Comparing error estimation measures for polynomial and kriging approximation of noise-free functions. Struct Multidiscip Optim 38(5):429–442. https://doi.org/10.1007/s00158-008-0290-z
6. Gu C, Wahba G (1991) Discussion: multivariate adaptive regression splines. Ann Stat 19(1):115–123. https://doi.org/10.1214/aos/1176347972
7. Clark DL, Bae HR, Gobal K, Penmetsa R (2016) Engineering design exploration using locally optimized covariance kriging. AIAA J 54(10):3160–3175. https://doi.org/10.2514/1.J054860
8. Yao W, Chen X, Zhao Y, Van Tooren M (2012) Concurrent subspace width optimization method for RBF neural network modeling. IEEE Trans Neural Netw Learn Syst 23(2):247–259. https://doi.org/10.1109/TNNLS.2011.2178560
9. Clarke SM, Griebsch JH, Simpson TW (2005) Analysis of support vector regression for approximation of complex engineering analyses. J Mech Des Trans ASME 127:1077–1087. https://doi.org/10.1115/1.1897403
10. Zhang J, Chowdhury S, Messac A (2012) An adaptive hybrid surrogate model. Struct Multidiscip Optim 46(2):223–238. https://doi.org/10.1007/s00158-012-0764-x
11. Yin H, Fang H, Wen G, Gutowski M, Xiao Y (2018) On the ensemble of metamodels with multiple regional optimized weight factors. Struct Multidiscip Optim 58(1):245–263. https://doi.org/10.1007/s00158-017-1891-1
12. Bhosekar A, Ierapetritou M (2018) Advances in surrogate based modeling, feasibility analysis, and optimization: A review. Comput Chem Eng 108:250–267. https://doi.org/10.1016/j.compchemeng.2017.09.017
13. Furnival GM, Wilson RW (2000) Regressions by leaps and bounds. Technometrics 42(1):69–79. https://doi.org/10.1080/00401706.2000.10485982
14. Tibshirani R (1996) Regression Shrinkage and Selection Via the Lasso. J Roy Stat Soc Ser B (Methodol) 58(1):267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x
15. Hosseinpour M, Sharifi H, Sharifi Y (2018) Stepwise regression modeling for compressive strength assessment of mortar containing metakaolin. Int J Model Simul 38(4):207–215. https://doi.org/10.1080/02286203.2017.1422096
16. O Giustolisi DS, Doglioni A (2004) Data Reconstruction and Forecasting By Evolutionary Polynomial Regression. 10.1142/9789812702838_0154
17. Zou H, Hastie T (2005) Regularization and variable selection via the elastic net. J R Stat Soc Ser B Stat Methodol 67(2):301–320. https://doi.org/10.1111/j.1467-9868.2005.00503.x
18. Goel T, Haftka RT, Shyy W, Queipo NV (2007) Ensemble of surrogates. Struct Multidiscip Optim 33(3):199–216. https://doi.org/10.1007/s00158-006-0051-9
19. Gu Y, Wei HL (2018) A robust model structure selection method for small sample size and multiple datasets problems. Inf Sci 451–452:195–209. https://doi.org/10.1016/j.ins.2018.04.007
20. Ong YS, Gupta A (2016) Evolutionary Multitasking: A Computer Science View of Cognitive Multitasking. Cogn Comput 8(2):125–142. https://doi.org/10.1007/s12559-016-9395-7
21. Naik A, Rangwala H (2018) Multi-task Learning. SpringerBriefs in Computer. Science 28(1):75–88
22. Fang J, Sun G, Qiu N, Kim NH, Li Q (2017) On design optimization for structural crashworthiness and its state of the art. Struct Multidiscip Optim 55(3):1091–1119. https://doi.org/10.1007/s00158-016-1579-y
23. Zhou XJ, Jiang T (2016) Metamodel selection based on stepwise regression. Struct Multidiscip Optim 54(3):641–657. https://doi.org/10.1007/s00158-016-1442-1
24. Fan C, Huang Y, Wang Q (2014) Sparsity-promoting polynomial response surface: A new surrogate model for response prediction. Adv Eng Softw 77:48–65. https://doi.org/10.1016/j.advengsoft.2014.08.001
25. Bischl B, Mersmann O, Trautmann H, Weihs C (2012) Resampling methods for meta-model validation with recommendations for evolutionary computation. Evol Comput 20(2):249–275
26. Simon R (2007). Resampling strategies for model assessment and selection. https://doi.org/10.1007/978-0-387-47509-7-8
27. Yanagihara H, Tonda T, Matsumoto C (2006) Bias correction of cross-validation criterion based on Kullback-Leibler information under a general condition. J Multivar Anal 97(9):1965–1975. https://doi.org/10.1016/j.jmva.2005.10.009
28. Lee H, Lee DJ, Kwon H (2018) Development of an optimized trend kriging model using regression analysis and selection process for optimal subset of basis functions. Aerosp Sci Technol 77:273–285. https://doi.org/10.1016/j.ast.2018.01.042
29. Jin Y, Wang H, Chugh T, Guo D, Miettinen K (2019) Data-driven evolutionary optimization: an overview and case studies. IEEE Trans Evol Comput 23(3):442–458. https://doi.org/10.1109/TEVC.2018.2869001
30. Liao P, Sun C, Zhang G, Jin Y (2020) Multi-surrogate multi-tasking optimization of expensive problems. Knowl-Based Syst 205. https://doi.org/10.1016/j.knosys.2020.106262
31. Ding J, Yang C, Jin Y, Chai T (2019) Generalized multitasking for evolutionary optimization of expensive problems. IEEE Trans Evol Comput 23(1):44–58. https://doi.org/10.1109/TEVC.2017.2785351
32. Wang H, Feng L, Jin Y, Doherty J (2021) Surrogate-assisted evolutionary multitasking for expensive minimax optimization in multiple scenarios. IEEE Comput Intell Mag 16(1):34–48. https://doi.org/10.1109/MCI.2020.3039067
33. Cheng MY, Gupta A, Ong YS, Ni ZW (2017) Coevolutionary multitasking for concurrent global optimization: with case studies in complex engineering design. Eng Appl Artif Intell 64:13–24. https://doi.org/10.1016/j.engappai.2017.05.008
34. Feng L, Zhou L, Zhong J, Gupta A, Ong YS, Tan KC, Qin AK (2019) Evolutionary multitasking via explicit autoencoding. IEEE Trans Cybern 49(9):3457–3470. https://doi.org/10.1109/TCYB.2018.2845361
35. Hurvich CM, Tsai CL (1989) Regression and time series model selection in small samples. Biometrika 76(2):297–307. https://doi.org/10.1093/biomet/76.2.297
36. Dette H, Pepelyshev A (2010) Generalized latin hypercube design for computer experiments. Technometrics 52(4):421–429. https://doi.org/10.1198/TECH.2010.09157
37. Jamil M, Yang XS (2013) A literature survey of benchmark functions for global optimisation problems. Int J Math Model Numer Optim 4(2):150–194. https://doi.org/10.1504/IJMMNO.2013.055204
38. Stickel JM, Nagarajan M (2012) Glass Fiber-Reinforced Composites: From Formulation to Application. Int J Appl Glas Sci 3(2):122–136. https://doi.org/10.1111/j.2041-1294.2012.00090.x
39. Kalnins K, Eglitis E, Jekabsons G, Rikards R (2008) Metamodels for Optimum Design of Laser Welded Sandwich Structures, pp 119–126. 10.1533/9781782420484.3.119

40. Gribonval R, Cevher V, Davies ME (2012) Compressible distributions for high-dimensional statistics. IEEE Trans Inf Theory 58(8):5016–5034. https://doi.org/10.1109/TIT.2012.2197174
41. Li P, Li H, Huang Y, Wang K, Xia N (2017) Quasi-sparse response surface constructing accurately and robustly for efficient simulation based optimization. Adv Eng Softw 114:325–336. https://doi.org/10.1016/j.advengsoft.2017.07.014

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