Struct Multidisc Optim
https://doi.org/10.1007/s00158-017-1807-0

RESEARCH PAPER

An adaptive RBF-HDMR modeling approach under limited computational budget

Haitao Liu[1] · Jaime-Rubio Hervas[2] · Yew-Soon Ong[2,3] · Jianfei Cai[2] · Yi Wang[4]

Received: 28 February 2017 / Revised: 3 June 2017 / Accepted: 3 September 2017

Abstract The metamodel-based high-dimensional model representation (e.g., RBF-HDMR) has recently been proven to be very promising for modeling high dimensional functions. A frequently encountered scenario in practical engineering problems is the need of building accurate models under limited computational budget. In this context, the original RBF-HDMR approach may be intractable due to the independent and successive treatment of the component functions, which translates in a lack of knowledge on when the modeling process will stop and how many points (simulations) it will cost. This article proposes an adaptive and tractable RBF-HDMR (ARBF-HDMR) modeling framework. Given a total of N_{max} points, it first uses N_{ini} points to build an initial RBF-HDMR model for capturing the characteristics of the target function f, and then keeps adaptively identifying, sampling and modeling the potential cuts with the remaining $N_{max} - N_{ini}$ points. For the second-order ARBF-HDMR, $N_{ini} \in [2n + 2, 2n^2 + 2]$ not only depends on the dimensionality n but also on the characteristics of f. Numerical results on nine cases with up to 30 dimensions reveal that the proposed approach provides more accurate predictions than the original RBF-HDMR with the same computational budget, and the version that uses the maximin sampling criterion and the best-model strategy is a recommended choice. Moreover, the second-order ARBF-HDMR model significantly outperforms the first-order model; however, if the computational budget is strictly limited (e.g., $2n + 1 < N_{max} \ll 2n^2 + 2$), the first-order model becomes a better choice. Finally, it is noteworthy that the proposed modeling framework can work with other metamodeling techniques.

Keywords Metamodeling · Adaptive high dimensional model representation · Limited computational budget · Tractable process

✉ Haitao Liu
htliu@ntu.edu.sg

Jaime-Rubio Hervas
jhervas@ntu.edu.sg

Yew-Soon Ong
ASYSOng@ntu.edu.sg

Jianfei Cai
ASJFCai@ntu.edu.sg

Yi Wang
Yi.Wang4@Rolls-Royce.com

[1] Rolls-Royce@NTU Corporate Laboratory, Nanyang Technological University, Singapore, 637460, Singapore

[2] School of Computer Science and Engineering, Nanyang Technological University, Singapore, 639798, Singapore

[3] Data Science and Artificial Intelligence Research Center, Nanyang Technological University, Singapore, 639798 Singapore

[4] Applied Technology Group, Rolls-Royce Singapore, 6 Seletar Aerospace Rise, Singapore, 797575 Singapore

1 Introduction

Metamodeling techniques have been extensively used in engineering design and optimization in order to alleviate computational costs. A complete review of typical metamodeling techniques, e.g., Kriging, radial basis functions (RBF), moving least square (MLS), support vector regression (SVM) and polynomial response (PR), can be found in Wang and Shan (2007) and Razavi et al. (2012). Although

Published online: 25 September 2017

being successful in various applications, it has been pointed out that in high dimensional scenarios, the accuracy and efficiency of these modeling techniques sharply decrease due to the "curse of dimensionality" (Shan and Wang 2010b). With the current increase of systems complexity, simulation-based engineering problems with large dimensionality (e.g., greater than 10) are frequently encountered. Therefore, a metamodeling technique that effectively tackles high dimensionality is needed.

One way to alleviate the "curse of dimensionality" is to approximate the target function by a combination of low-dimensional functions. In this context, Shan and Wang (2010b) reviewed and classified the strategy into two categories: adaptive computation and additive models. Examples of adaptive computation include projection pursuit regression (Friedman and Stuetzle 1981) and classification and regression trees (CART) (Breiman et al. 1984); examples of additive models include additive interactive regression (Andrews and Whang 1990) and high-dimensional model representation (HDMR). Here, we mainly focus on the HDMR strategy.

The well-known HDMR strategy is developed from some seminal works by (Sobol 1993, 2003; Rabitz et al. 1999). It follows the theorem that for any integrable function, there exists a unique high-dimensional model representation that is of finite order and offers a hierarchical structure. In a similar spirit to Taylor expansion, the HDMR contains a family of representations in which each reflects the individual and correlated contributions of the input variables to the output response. Thus, it can act as a general set of quantitative model assessment and analysis tools. Thereafter, a family of HDMRs with different characteristics was proposed for the purpose of, for instance, sensitivity analysis, reliability analysis and modeling (Rabitz and Ališ 1999, Li et al. 2001a, b, 2002, 2006, 2008; Tunga and Demiralp 2005, Chowdhury and Rao 2009; Liu et al. 2016d).

Among these HDMRs, there are mainly two expansions, namely, ANOVA-HDMR and Cut-HDMR. In terms of sensitivity analysis, the ANOVA-HDMR (Rabitz and Ališ 1999, Li et al. 2002, 2006, 2008) is more suitable since it is designed for statistical purposes and is capable of identifying important variables and correlations. The main drawback of ANOVA-HDMR is that it needs to compute many integrals. On the other hand, the Cut-HDMR (Li et al. 2001a, b), which involves no integral operations, is an exact representation of the target function $f(\mathbf{x})$ by decomposing $f(\mathbf{x})$ into a set of component functions on *cut lines*, *cut planes* and *cut hyperplanes* passing through a user-defined *cut point* (also known as reference point). Therefore, in terms of modeling, the Cut-HDMR is more attractive due to the ease of implementation. The original Cut-HDMR, however, is incomplete since it only offers a check-up table.

Shan and Wang (2010a) combined RBF with Cut-HDMR to construct a promising high-dimensional modeling approach, termed RBF-HDMR. The schematic flowchart of the second-order RBF-HDMR is shown in Fig. 1. The RBF-HDMR framework approximates the component functions on the cut lines and planes successively via RBF and an accompanying sampling strategy, and then combines all the component RBF models together to form a complete model. In this metamodel based HDMR modeling framework, different effective sampling approaches, e.g., maximin sampling (Ulaganathan et al. 2016), DIRECT-based sampling (Huang et al. 2015) and LOLA-Voronoi sampling (Ulaganathan et al. 2016; Cai et al. 2016), can be used for the accompanying sampling process; similarly, different metamodeling techniques, e.g., Kriging (Tang et al. 2013), gradient-based Kriging (Ulaganathan et al. 2016), enhanced RBF model (Cai et al. 2016), MLS (Li et al. 2012) and SVR (Huang et al. 2015), can be employed for approximating the component functions.

These metamodel-based HDMR approaches are promising for high-dimensional modeling; however, they present two important disadvantages when handling practical engineering problems, namely:

- The model construction cannot utilize *random* points. As shown in Fig. 1, the typical HDMR modeling process requires *well-structured* points sampled only on cut lines and planes. In practice, however, cases where there exists some data lying outside the cut lines and planes are frequently encountered. To address this issue, Liu et al. (2017b) proposed a generalized RBF-HDMR (GRBF-HDMR) model by using the error-allocation strategy to utilize random data;
- The model construction is *passive* and *intractable*. As shown in Fig. 1, current metamodel-based HDMRs *locally* approximate the component functions, i.e., they are modeled independently and successively until convergence. It is intractable since there is a lack of knowledge on when the process will stop and how many points it will cost. It can be argued that the modeling process can be forcibly terminated when the computational budget is exhausted. But this will result in an incomplete model, since some component functions may not have been approximated yet. Besides, due to the local view, the over-/under-sampling phenomenon often occurs on some cuts, and deteriorates the prediction quality.

This article focuses on the second issue, and presents an *adaptive* and *tractable* RBF-HDMR (ARBF-HDMR) approach to build an accurate model under limited computational budget, e.g., given N_{max} points (simulations). The proposed approach adaptively refines the model in a tractable and open way: it first builds an initial RBF-HDMR

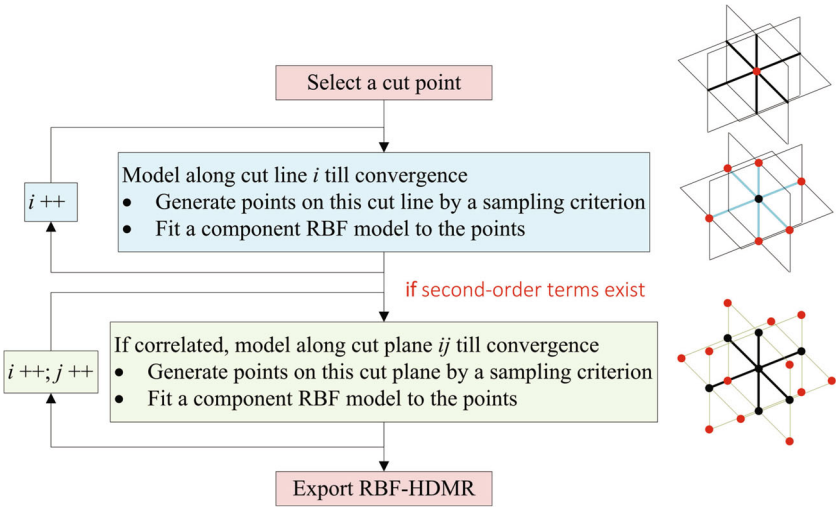

Fig. 1 Schematic flowchart of the second-order RBF-HDMR modeling process

model to have a *global view* on the characteristics of the target function, and then adaptively identifies, samples and models the *potential* cuts until the computational budget is exhausted.

The remaining of this article is organized as follows. Section 2 introduces the original RBF-HDMR modeling approach. Section 3 presents the proposed ARBF-HDMR modeling approach. Sections 4 and 5 assess the performance of ARBF-HDMR through extensive numerical experiments. Finally, Section 6 provides some concluding remarks.

2 The original RBF-HDMR modeling approach

2.1 The RBF interpolation model

The RBF interpolation model is a family of models with the same structure but distinguished by the choice of basis functions (Fang and Horstemeyer 2006). Suppose that $f(\mathbf{x})$ is the target function to be approximated in the design space $D \in \mathbb{R}^n$. Given a set of observed points $\mathbf{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_m\}^T$ and their output responses $\mathbf{y} = \{f(\mathbf{x}_1), \cdots, f(\mathbf{x}_m)\}^T$, the RBF model $\hat{f}(\mathbf{x})$ is

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{m} \lambda_i \phi\left(\|\mathbf{x} - \mathbf{x}_i\|\right) + \sum_{j=1}^{u} \mu_j p_j(\mathbf{x}) \tag{1}$$

where ϕ is a basis function, λ_i is the weight for the i-th observed point, p is a linear polynomial function, μ_j is the coefficient for the j-th polynomial term, and u is the number of polynomials. A widely used basis function has the multi-quadric form

$$\phi(r) = \sqrt{(r^2 + s^2)} \tag{2}$$

where $r = \|\mathbf{x} - \mathbf{x}_i\|$ is the Euclidean distance between two points, and $s > 0$ is a user-defined shape parameter. Other forms of ϕ such as the Gaussian basis function $\phi(r) = e^{-r^2/s^2}$ and the cubic basis function $\phi(r) = (r+s)^3$ may also be used.

To determine the coefficients $\boldsymbol{\lambda} = \{\lambda_1, \cdots, \lambda_m\}^T$ and $\boldsymbol{\mu} = \{\mu_1, \cdots, \mu_u\}^T$, we adopt the interpolation conditions

$$\hat{f}(\mathbf{x}_i) = f(\mathbf{x}_i), \quad 1 \le i \le m \tag{3}$$

and the additional orthogonality conditions

$$\sum_{i=1}^{m} \lambda_i p_j(\mathbf{x}_i) = 0, \quad 1 \le j \le u \tag{4}$$

Thereafter, by defining $A_{ik} = \phi(\|\mathbf{x}_i - \mathbf{x}_k\|)$ $(1 \le i, k \le m)$ and $P_{ij} = p_j(\mathbf{x}_i)$ $(1 \le i \le m, 1 \le j \le u)$, the RBF model is obtained by solving the following $(m+u) \times (m+u)$ linear system

$$\begin{pmatrix} \mathbf{A} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\mu} \end{pmatrix} = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} \tag{5}$$

The RBF model is found to be simple and effective for approximating complex functions from low to high dimensions (Liu et al. 2016b).

2.2 The RBF-HDMR model for high-dimensional nonlinear problems

Let $\mathbf{x} = (x_1, \cdots, x_n) \in [0,1]^n$ be an n-dimensional point, a general HDMR model for the target function $f(\mathbf{x})$ can be expressed in a hierarchical structure as (Rabitz and Aliş 1999)

$$\begin{aligned} f(\mathbf{x}) = {} & f_0 + \sum_{i=1}^{n} f_i(x_i) + \sum_{1 \le i < j \le n} f_{ij}(x_i, x_j) + \cdots \\ & + \sum_{1 \le i_1 < \cdots < i_l \le n} f_{i_1 \cdots i_l}(x_{i_1}, \cdots, x_{i_l}) \\ & + f_{1 \cdots n}(x_1, \cdots, x_n) \end{aligned} \tag{6}$$

where the constant term f_0 denotes the zeroth-order effect on $f(\mathbf{x})$, the first-order term $f_i(x_i)$ is the effect of the single variable x_i on $f(\mathbf{x})$, the second-order term $f_{ij}(x_i, x_j)$ is the correlated contribution of the two variables x_i and x_j on $f(\mathbf{x})$ after removing their individual effects, the higher-order terms reflect the effects of the increasing numbers of correlated variables acting together on $f(\mathbf{x})$, and the final term $f_{1\cdots n}(x_1, \cdots, x_n)$ denotes the residual effect of all the variables acting together on $f(\mathbf{x})$ after all the lower-order correlations and individual effects have been removed.

For an n-dimensional target function, the number of all possible component functions in a complete HDMR expansion is

$$N = \sum_{i=0}^{n} \frac{n!}{(n-i)!i!} = 2^n \tag{7}$$

and the expansion is always exact (Rabitz et al. 1999). It is observed that N dramatically increases with the dimensionality. Fortunately, for most physical systems, only the low-order correlations among variables significantly influence the output response (Rabitz and Aliş 1999). In this context, the second-order HDMR expansion is widely used given that it provides an acceptable modeling accuracy for most physical systems while significantly reducing the required computing cost.

Among existing HDMRs (Li et al. 2001b; Tunga and Demiralp 2005; Wang et al. 2003), the Cut-HDMR (Rabitz and Aliş 1999) is found to have a simple structure and provide a cost-efficient model with a similar accuracy to other HDMRs. Hence, the Cut-HDMR is chosen as the basis for RBF-HDMR (Shan and Wang 2010a). The Cut-HDMR expansion approximates along the cuts (cut lines, planes and hyperplanes) passing through a cut point in the domain. By choosing a cut point $\mathbf{x}_0 = (x_{1_0}, \cdots, x_{n_0})$, the expressions of the component functions in Cut-HDMR are given as

$$f_0 = f(\mathbf{x}_0) \tag{8}$$

$$f_i(x_i) = f(x_i, \mathbf{x}_0^i) - f_0 \tag{9}$$

$$f_{ij}(x_i, x_j) = f(x_i, x_j, \mathbf{x}_0^{ij}) - f_i(x_i) - f_j(x_j) - f_0 \tag{10}$$

$$f_{ijk}(x_i, x_j, x_k) = f(x_i, x_j, x_k, \mathbf{x}_0^{ijk}) - f_{ij}(x_i, x_j) - f_{ik}(x_i, x_k) - f_{jk}(x_j, x_k) - f_i(x_i) - f_j(x_j) - f_k(x_k) - f_0 \tag{11}$$

$$\vdots$$

$$f_{1\cdots n}(x_1, \cdots, x_n) = f(\mathbf{x}) - f_0 - \sum_i f_i(x_i) - \sum_{ij} f_{ij}(x_i, x_j) - \cdots \tag{12}$$

where $\mathbf{x}_0^i$, $\mathbf{x}_0^{ij}$, and $\mathbf{x}_0^{ijk}$ denote $\mathbf{x}_0$ without the elements x_{i_0}; x_{i_0}, x_{j_0}; and x_{i_0}, x_{j_0}, x_{k_0}; respectively. The points $\mathbf{x}_0$, $(x_i, \mathbf{x}_0^i) = (x_{1_0}, \cdots, x_i, \cdots, x_{n_0})$, and $(x_i, x_j, \mathbf{x}_0^{ij}) = (x_{1_0}, \cdots, x_i, \cdots, x_j, \cdots, x_{n_0})$, that lie on cut lines and planes, are referred as the zeroth-, first- and second-order points, respectively. Correspondingly, $f(x_i, \mathbf{x}_0^i)$ is the response of $f(\mathbf{x})$ at $(x_i, \mathbf{x}_0^i)$, and $f(x_i, x_j, \mathbf{x}_0^{ij})$ is the response of $f(\mathbf{x})$ at $(x_i, x_j, \mathbf{x}_0^{ij})$. While $f_i(x_i)$, which is different from $f(x_i, \mathbf{x}_0^i)$, is the first-order component function response at x_i along the i-th cut line, and $f_{ij}(x_i, x_j)$ is the second-order component function response at (x_i, x_j) on the i-j cut plane.

Equations (8)–(12) only offer a check-up table, and cannot be used for data interpolation in this current version. To have a complete and available model, Shan and Wang (2010a) proposed to use the RBF technique together with an accompanying sampling strategy to approximate the component functions.

Some notations are defined before introducing the RBF-HDMR model. The point set $\mathbf{X}_i = \{(x_i, \mathbf{x}_0^i)\}^T$ contains the first-order points on the i-th cut line, the vector $\mathbb{X}_i = \{x_i\}^T$ is the i-th column of $\mathbf{X}_i$, and $\mathbf{Y}_i = \{f_i(x_i)\}^T$ is a vector of the first-order component function responses calculated by Eq. (9). Similarly, the set $\mathbf{X}_{ij} = \{(x_i, x_j, \mathbf{x}_0^{ij})\}^T$ collects the second-order points on the i-j cut plane, the set $\mathbb{X}_{ij} = \{(x_i, x_j)\}^T$ picks the i-th and j-th columns of $\mathbf{X}_{ij}$, and $\mathbf{Y}_{ij} = \{f_{ij}(x_i, x_j)\}^T$ contains the second-order component function responses computed by (10).

The second-order RBF-HDMR model, which is sufficiently accurate for most of the practical cases, is expressed as

$$\begin{aligned} f(\mathbf{x}) &\approx f_0 + \sum_{i=1}^{n} f_i(x_i) + \sum_{1\le i<j\le n} f_{ij}(x_i, x_j) \\ &\approx f_0 + \sum_{i=1}^{n} \hat{f}_i(x_i) + \sum_{1\le i<j\le n} \hat{f}_{ij}(x_i, x_j) = \hat{f}(\mathbf{x}) \end{aligned} \tag{13}$$

where $\hat{f}_i$, a 1D RBF model built using $\mathbb{X}_i$ and $\mathbf{Y}_i$, is the approximation of the first-order component function f_i; and $\hat{f}_{ij}$, a 2D RBF model built using $\mathbb{X}_{ij}$ and $\mathbf{Y}_{ij}$, is the approximation of the second-order component function f_{ij}. The second-order RBF-HDMR modeling process is conceptually illustrated in Fig. 1. It first approximates the first cut line until convergence, and then the remaining cut lines successively. If the second-order terms exist, it again approximates the cut planes successively until convergence. The convergence of the RBF modeling process on each cut is solely measured by the relative prediction error at a test point.

The RBF-HDMR, which turns a black-box function into an analytical functional form, presents a modeling framework which reveals the nonlinearity of the function with respect to the single variables, the existence of high-order terms, and the correlation of two variables in order to reduce

the required number of points for modeling. Moreover, this information can further help RBF-HDMR to perform well for sensitivity analysis, decomposition, visualization, optimization, etc. It is to be noted that Shan and Wang (2011) efficiently extended the modeling process of RBF-HDMR from the second-order form to any higher order.

In this article, unless otherwise indicated, the RBF-HDMR model and the proposed ARBF-HDMR model are constructed in the second-order fashion.

3 An adaptive RBF-HDMR modeling approach

As shown in Fig. 2, the proposed ARBF-HDMR modeling approach is composed of three steps: (1) building an initial RBF-HDMR model to capture the characteristics of the target function; (2) identifying a potential cut; and (3) sampling a new point and refining the RBF model on the potential cut. Steps (2) and (3) form an adaptive and tractable modeling framework that (a) adaptively and efficiently improves the prediction quality of the ARBF-HDMR model; and (b) allows to stop and export a complete model at any time.

3.1 Construction of an initial RBF-HDMR model

The first step is to build an initial RBF-HDMR model for having a global view on the characteristics of the target function, e.g., the linearity of the first-order component functions, the existence of the second-order terms, and the correlations between any two variables. The global view of the target function can help removing redundant terms, thus (1) saving computing resources and (2) making the subsequent modeling process more targeted. The initial RBF-HDMR model is suggested to be built with as few points as possible such that the majority of the subsequent points are adaptively sampled, which is beneficial for the final model accuracy.

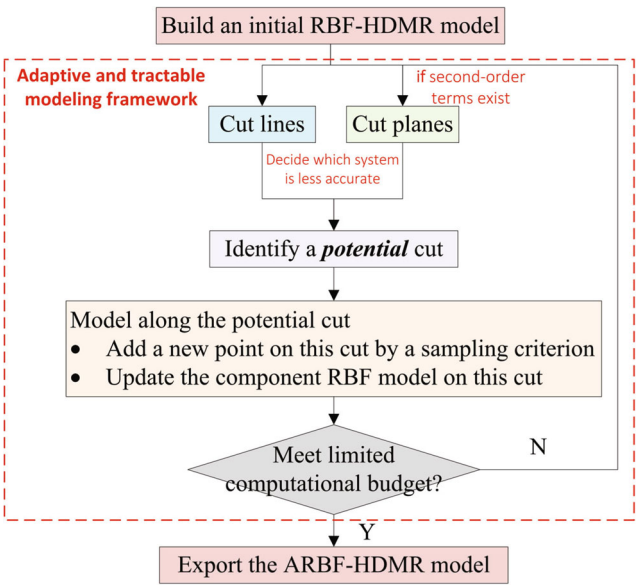

Fig. 2 Schematic flowchart of the ARBF-HDMR modeling process

To this end, the initial RBF-HDMR model for the target function $f(\mathbf{x})$ in $D \in [0, 1]^n$ is constructed in a regular way as follows:

1. The center point of the domain D is selected as the cut point $\mathbf{x}_0$ wherein $x_{i_0} = 0.5$;
2. Two end points $(x_i^+, \mathbf{x}_0^i)$ and $(x_i^-, \mathbf{x}_0^i)$ along the i-th cut line are added to fit the first-order RBF model $\hat{f}_i$. Note that, in the unit space, we have $x_i^+ = 1$ and $x_i^- = 0$;
3. After building the first-order RBF-HDMR model $\hat{f}(\mathbf{x}) = f_0 + \sum_{i=1}^n \hat{f}_i(x_i)$, a corner point $(x_1^+, \cdots, x_n^+)$ is used to check for the existence of second-order terms. That is, if the relative error of $\hat{f}$ (i.e., $|(\hat{f} - f)/f|$) at this validation point is less than 0.1, the modeling process terminates; otherwise, it goes to step 4;
4. In the $i - j$ cut plane, a validation point $(x_i^+, x_j^+, \mathbf{x}_0^{ij})$ is used to check for the correlation between the variables x_i and x_j. If they are correlated, i.e., the relative error of $\hat{f}$ is less than 0.1 at the validation point, four corner points are added on the i-j cut plane to fit the second-order RBF model $\hat{f}_{ij}$.
5. Export the initial RBF-HDMR model $\hat{f}(\mathbf{x}) = f_0 + \sum_{i=1}^n \hat{f}_i(x_i) + \sum_{1 \le i < j \le n} \hat{f}_{ij}(x_i, x_j)$.

Algorithm 1 provides a detailed description of the initial RBF-HDMR modeling process.

It is found that for an n-dimensional target function, the number of points N_{ini} required to build the initial RBF-HDMR model satisfies

$$N_{ini} \le 1 + 2n + 1 + \frac{4n(n-1)}{2} = 2n^2 + 2 \tag{14}$$

where the first term "1" represents the cut point in step 1, the second term "$2n$" represents the number of points sampled on n cut lines in step 2, the third term "1" represents the validation point to check for the existence of second-order terms in step 3, and the final term "$4n(n-1)/2$" represents the number of points sampled on the cut planes if all the input variables are correlated in step 4.

It is worth noting that the N_{ini} value for the second-order ARBF-HDMR not only depends on the dimensionality but also on the characteristics of the target function $f(\mathbf{x})$, e.g., the existence of second-order terms, and the correlations among input variables. If $f(\mathbf{x})$ is relatively simple and has no second-order terms, then $N_{ini} = 1 + 2n + 1 = 2n + 2$; if $f(\mathbf{x})$ is very complex, i.e., it exists second-order terms and any two variables are correlated, then $N_{ini} = 2n^2 + 2$; otherwise, we have $2n + 2 < N_{ini} < 2n^2 + 2$. In practice, since $f(\mathbf{x})$ is a black-box function, N_{ini} cannot be

determined in advance. Equation (14) indicates that we need at most $2n^2+2$ points to accomplish the initial RBF-HDMR modeling process. Therefore, to build a complete second-order ARBF-HDMR model, the allowed maximal number of points N_{max} should be greater than $2n^2+2$. Besides, as will be discussed in Section 5.4.1, if the computational budget is strictly limited, e.g., N_{max} is much less than $2n^2+2$, we prefer building the first-order ARBF-HDMR wherein $N_{ini} = 1+2n$ due to the ignorance of the second-order terms.

Algorithm 1 Construction of the initial RBF-HDMR model

Input: A target function f defined in the domain $D \in \mathbb{R}^n$
Output: An initial RBF-HDMR model $\hat{f}$
begin:
Select the center point of the domain to be the cut point $\mathbf{x}_0$ and evaluate its response as f_0;
for $i = 1 : n$
Add two end points $(x_i^+, \mathbf{x}_0^i)$ and $(x_i^-, \mathbf{x}_0^i)$ on the i-th cut line and obtain their first-order responses $f_i(x_i^+)$ and $f_i(x_i^-)$ by (9);
Update the sample set $\mathbb{X}_i$ and the first-order response set $\mathbf{Y}_i$, and construct the i-th first-order RBF model $\hat{f}_i \leftarrow \mathbb{X}_i, \mathbf{Y}_i$.
end
Form a validation point $(x_1^+, \cdots, x_n^+)$ to check for the existence of second-order terms.
if The second-order terms exist
for $i = 1 : n$, $j = i+1 : n$
Form a validation point $(x_i^+, x_j^+, \mathbf{x}_0^{ij})$ to check for the correlation of variables x_i and x_j;
if x_i and x_j are correlated
Add four corner points $(x_i^+, x_j^+, \mathbf{x}_0^{ij})$, $(x_i^+, x_j^-, \mathbf{x}_0^{ij})$, $(x_i^-, x_j^+, \mathbf{x}_0^{ij})$ and $(x_i^-, x_j^-, \mathbf{x}_0^{ij})$ on the i-j cut plane, and obtain their second-order responses $f_{ij}(x_i^+, x_j^+, \mathbf{x}_0^{ij})$, $f_{ij}(x_i^+, x_j^-, \mathbf{x}_0^{ij})$, $f_{ij}(x_i^-, x_j^+, \mathbf{x}_0^{ij})$ and $f_{ij}(x_i^-, x_j^-, \mathbf{x}_0^{ij})$ by (10);
Update the sample set $\mathbb{X}_{ij}$ and the second-order response set $\mathbf{Y}_{ij}$, and construct the second-order RBF model $\hat{f}_{ij} \leftarrow \mathbb{X}_{ij}, \mathbf{Y}_{ij}$.
end
end
end
Export the initial RBF-HDMR model $\hat{f}(\mathbf{x}) = f_0 + \sum_{i=1}^n \hat{f}_i(x_i) + \sum_{1\le i<j\le n} \hat{f}_{ij}(x_i, x_j)$.
end

3.2 Identification of potential cuts

After building the initial RBF-HDMR model, the second step is to identify potential cuts for the subsequent adaptive modeling process. A cut is said to be potential if the associated RBF model has large prediction errors or large prediction uncertainty. Thus, improving the RBF model on the potential cut is beneficial for the improvement of the overall ARBF-HDMR model.

The identification process operates in two levels: (1) at the *system level*, it identifies a potential system that has a greater impact on the overall accuracy of the RBF-HDMR model between the first-order RBF model system $\mathcal{F}_I = \{\hat{f}_i\}_{1\le i\le n}$ and the second-order RBF model system $\mathcal{F}_{II} = \{\hat{f}_{ij}\}_{1\le i<j\le n}$; (2) at the *cut level*, it identifies the potential cut line (plane) on which the RBF model has the worst performance among the identified model system.

Note that the identification is first conducted at the system level by treating the model systems $\mathcal{F}_I$ and $\mathcal{F}_{II}$ separately. This is because: (1) these two kinds of RBF models usually have considerable differences in magnitude in terms of predictions, since the second-order functions play a residual role after removing the impact of the first-order functions; and (2) they have different dimensions and characteristics.

To assist the identification process, we first need to estimate the accuracies of the RBF models. Here, a well-known generalized cross validation mean square error (GMSE) criterion, also known as leave-one-out (LOO) cross validation error criterion, is employed to assess the model accuracy as

$$\text{GMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m} e_{LOO}^2(\mathbf{x}_i)} \tag{15}$$

where

$$e_{LOO}(\mathbf{x}_i) = f(\mathbf{x}_i) - \hat{f}^{-i}(\mathbf{x}_i) \tag{16}$$

In (16), $\hat{f}^{-i}$ is a RBF model built using all the sampled points except the i-th one. An accurate RBF model prefers a small GMSE value. The GMSE value usually overestimates the real model error but is capable of successfully estimating the real model error with sufficient points (Viana et al. 2009; Liu et al. 2016c). Using (15), the global error of $\hat{f}_i$ is estimated as GMSE_i, and the global error of $\hat{f}_{ij}$ is as GMSE_{ij}. Given m sampled points, the RBF modeling process augmented with u polynomial terms is equivalent to solving a $(m+u)\times(m+u)$ linear system, see (5). To obtain the GMSE value by (15), we should remove one of the m sampled points each time and refit the RBF model. That is, we need to solve a $(m+u-2)\times(m+u-2)$ linear system m times. The LU decomposition can be used to solve the linear system; however, the computational complexity associated with the GMSE calculation using this approach is $\mathcal{O}(m^4)$, which is especially time-consuming for cases with large sample sizes and high dimensions. For the original RBF model without polynomial terms, Rippa (1999) presented a simple formula to speed up the LOO computation in (16). Here, for the RBF

model with polynomial terms, we derive a similar formula for the fast evaluation of the LOO error.

Let the linear system of RBF be organized as

$$\mathbf{B}\boldsymbol{c} = \mathbf{f} \tag{17}$$

where

$$\mathbf{B} = \begin{pmatrix} \mathbf{A} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{pmatrix}, \boldsymbol{c} = \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\mu} \end{pmatrix}, \mathbf{f} = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix}$$

Then, the LOO error at the observed point $\mathbf{x}_i$ $(1 \le i \le m)$ can be calculated as

$$e_{LOO}(\mathbf{x}_i) = f(\mathbf{x}_i) - \hat{f}^{-i}(\mathbf{x}_i) = \frac{c_i}{B_{ii}^{-1}} \tag{18}$$

where c_i is the i-th element of the interpolation coefficient vector $\boldsymbol{c}$, and B_{ii}^{-1} is the i-th diagonal element of the inverse of the interpolation matrix $\mathbf{B}$. For calculating the LOO errors at m sampled points, we need to compute the inverse of the matrix $\mathbf{B}$ only once, which results in the computational complexity of $\mathcal{O}(m^3)$.

At the system level, after obtaining the GMSE values of the RBF models, according to the short board effect, the overall accuracy of system $\mathcal{F}_I$ or $\mathcal{F}_{II}$ is represented by the accuracy of the worst member. That is,

$$\begin{cases} \text{GMSE}_I = \max\{\text{GMSE}_i\}, & 1 \le i \le n \\ \text{GMSE}_{II} = \max\{\text{GMSE}_{ij}\}, & 1 \le i < j \le n \end{cases} \tag{19}$$

If $\text{GMSE}_I > \text{GMSE}_{II}$, the first-order model system $\mathcal{F}_I$ is said to be the potential system, since the less accurate $\mathcal{F}_I$ has a greater impact on the overall accuracy of the ARBF-HDMR. Hence, a new point is added on one of the cut lines to quickly improve the overall accuracy of the ARBF-HDMR. Oppositely, if $\text{GMSE}_I < \text{GMSE}_{II}$, then the second-order residual model system $\mathcal{F}_{II}$ has a greater impact on the overall accuracy of the ARBF-HDMR. Consequently, a new point is added on one of the correlated cut planes.

Next, at the cut level, the potential cut among the potential system is identified. Assume that $\text{GMSE}_I > \text{GMSE}_{II}$, then a score function is employed to rank the cut lines as

$$\mathcal{S}_i = \frac{\text{GMSE}_i}{\sum_i \text{GMSE}_i} + \frac{\rho_i}{\sum_i \rho_i}, \quad 1 \le i \le n \tag{20}$$

where the first term of the right-hand side is the normalized GMSE value of $\hat{f}_i$, and $\rho_i = 1/m_i$ in the second term is the sample density on the i-th cut line where there are m_i first-order points. It is found that the first term is a local exploitation term that draws attention to the cut lines where the RBF models have larger prediction errors; whereas the second term is a global exploration term over the system that draws attention to the cut lines with a lower sample density (i.e., higher uncertainty). The global exploration term

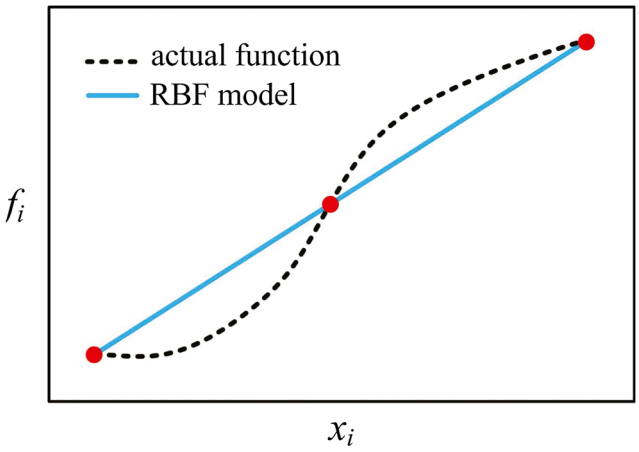

Fig. 3 A case showing the necessity of the global exploration term

can help avoid missing some undetected features on a cut line. For instance, Fig. 3 illustrates a case where the original RBF-HDMR process asserts that this cut line has passed the linearity check and will not visit it again. Actually, the main characteristics of the output response along this line have not been revealed yet. By considering the global exploration term, this kind of cut line can be revisited.

Similarly, if $\text{GMSE}_{II} > \text{GMSE}_I$, the score of the i-j cut plane can be calculated as

$$\mathcal{S}_{ij} = \frac{\text{GMSE}_{ij}}{\sum_{i,j} \text{GMSE}_{ij}} + \frac{\rho_{ij}}{\sum_{i,j} \rho_{ij}}, \quad 1 \le i < j \le n \tag{21}$$

where the sample density on the i-j cut plane with m_{ij} points is as $\rho_{ij} = 1/m_{ij}$.

3.3 Effective sampling and modeling on potential cuts

After obtaining a potential cut, a new point should be added to refit the RBF model on it. To further improve the prediction quality, this section investigates effective sampling and modeling strategies on potential cuts.

3.3.1 Sampling on potential cuts

Given that the ARBF-HDMR modeling process is conducted in a sequential manner, a sequential sampling approach is recommended to sample a new point $\mathbf{x}_{new}$ on the potential cut. Current sequential sampling approaches can be classified into two categories: space-filling sequential sampling strategy (Johnson et al. 1990; Crombecq et al. 2011b; Liu et al. 2015) which generates points to fill the entire domain evenly, and adaptive sequential sampling strategy (Crombecq et al. 2011a; Xu et al. 2014; Liu et al. 2016a, 2017a) which purposefully generates more points in regions where the model yields large prediction errors.

Among the space-filling sequential sampling approaches, the well-known maximin sampling approach (Johnson et al. 1990) is employed. This approach selects a new point as the one that lies farthest away from the m existing points. That is,

$$\mathbf{x}_{new} = arg \max_{\mathbf{x}} \min_{i} \|\mathbf{x} - \mathbf{x}_i\|, \quad 1 \le i \le m \tag{22}$$

where $\mathbf{x}_i$ is an existing point on the potential cut.

Among the adaptive sequential sampling approaches, the simple and effective CV-Voronoi sampling approach (Xu et al. 2014) is introduced. The CV-Voronoi first employs the Voronoi diagram algorithm to partition the entire domain into a set of Voronoi cells as

$$C_i = \{\mathbf{x} \in D, | \ \|\mathbf{x} - \mathbf{x}_i\| \le \|\mathbf{x} - \mathbf{x}_j\|\}, \quad 1 \le i \ne j \le m \tag{23}$$

Thereafter, the cross-validation approach is adopted to estimate the prediction error of each cell, and the one with the largest error is denoted as the sensitive cell

$$C_{sensitive} = arg \max_{C_i} e(C_i) \tag{24}$$

where $e(C_i) = |e_{LOO}(\mathbf{x}_i)| = | \ f(\mathbf{x}_i) - \hat{f}^{-i}(\mathbf{x}_i) \ |$. Finally, the new point $\mathbf{x}_i$ is selected in the sensitive cell as

$$\mathbf{x}_{new} = arg \max_{\mathbf{x} \in C_{sensitive}} \|\mathbf{x} - \mathbf{x}_{sensitive}\| \tag{25}$$

To investigate which kind of sampling strategy is more effective for the ARBF-HDMR, the maximin sampling approach and the CV-Voronoi sampling approach are used in this article.

3.3.2 Modeling on potential cuts

It is observed that the quality of the component RBF models has a great impact on the performance of the final ARBF-HDMR model. It is known that the RBF models with different basis functions yield different predictions, and the component functions in HDMR may have different characteristics. Hence, rather than building RBF models with a fixed basis function for all the component functions, we employ a best-model strategy to select the best RBF model with an appropriate basis function for each component function.

For a component function, K RBF models using K different basis functions are first constructed to form a pool of RBF models as

$$\mathcal{F} = \{\hat{f}^1_{\phi_1}, \cdots, \hat{f}^K_{\phi_K}\} \tag{26}$$

The best-model strategy uses the GMSE criterion to evaluate the accuracies of the K RBF model, and selects the one with the smallest GMSE

$$\hat{f}^b(x) = \hat{f}^u(x) \in \mathcal{F} \tag{27}$$

where $u = arg \min_{1 \le u \le K} \text{GMSE}_u$.

Through the best-model strategy, the component functions with different characteristics may be separately modeled by the RBF models with different basis functions, thus yielding more accurate predictions. Note that the best-model strategy is employed to build the RBF models in both the initial RBF-HDMR process and the adaptive modeling process.

3.4 Description of the ARBF-HDMR approach

For the target function, after building an initial RBF-HDMR model, Algorithm 2 elaborates on the subsequent ARBF-HDMR modeling process under limited computational budget. Compared to the passive and intractable RBF-HDMR framework in Fig. 1, it is found that the proposed modeling framework is adaptive and tractable, since

- it keeps monitoring the potential cut on which the RBF model has the poorest performance, thus effectively improving the overall model accuracy;
- in addition, the modeling framework runs in an open way so that it can stop and export a complete model at any time, which is suitable for practical problems with limited computational budget.

In Algorithm 2, the sampling criterion could be either the maximin sampling criterion or the CV-Voronoi sampling criterion. Both of the two criteria obtain new points by solving an auxiliary optimization problem (Johnson et al. 1990; Xu et al. 2014). It is found that for ARBF-HDMR, the cuts simply lie in a 1D or 2D space. Hence, to query the best new point using the two sampling criteria, the auxiliary optimization problem is solved based on 1000 uniform candidate points on a cut line, and 100 $\times$ 100 grid candidate points on a cut plane.

In the numerical experiments below, several versions of the ARBF-HDMR are adopted. The basic version, denoted as ARBF$_{MM}$-HDMR, uses the maximin sampling approach to add new points on potential cuts and the RBF model with multi-quadric basis function to approximate the component functions. The version that adopts the CV-Voronoi sampling approach and the RBF model with multi-quadric basis function is denoted as ARBF$_{CVV}$-HDMR. Finally, the version that adopts the maximin sampling approach and the best-model strategy is denoted as ARBF$^b_{MM}$-HDMR.

Algorithm 2 Construction of the ARBF-HDMR model

Input: A target function f defined in the domain $D \in \mathbb{R}^n$; an initial RBF-HMDR model $\hat{f}$ constructed with N_{ini} points; the maximal number of function evaluations $N_{max} > N_{ini}$

Output: An ARBF-HDMR model $\hat{f}_A$

begin:

for $i = 1 : N_{max} - N_{ini}$

Use (19) to measure the accuracy GMSE_I of the first-order RBF model system $\mathcal{F}_I$ and, if the second-order terms exist, measure the accuracy GMSE_{II} of the second-order RBF model system $\mathcal{F}_{II}$;

if $\text{GMSE}_I \geq \text{GMSE}_{II} \parallel$ there are no second-order terms

Use the score function (20) to identify a potential cut line, and then select a new point x_i^{new} from 1000 evenly distributed candidate points by a sampling criterion on this cut line;

Evaluate the new point to obtain its first-order response $f_i(x_i^{new})$;

Update the sample set $\mathbb{X}_i$ and the first-order response set $\mathbf{Y}_i$, and reconstruct the i-th first-order RBF model $\hat{f}_i \leftarrow \mathbb{X}_i, \mathbf{Y}_i$.

elseif $\text{GMSE}_I < \text{GMSE}_{II}$

Use the score function (21) to identify a potential cut plane, and select a new point (x_i^{new}, x_j^{new}) from 100×100 grid candidate points by a sampling criterion on this cut plane;

Evaluate the new point to obtain its second-order response $f_{ij}(x_i^{new}, x_j^{new})$;

Update the sample set $\mathbb{X}_{ij}$ and the second-order response set $\mathbf{Y}_{ij}$, and reconstruct the second-order RBF model $\hat{f}_{ij} \leftarrow \mathbb{X}_{ij}, \mathbf{Y}_{ij}$.

end

end

Export the final ARBF-HDMR model $\hat{f}_A(\mathbf{x}) = f_0 + \sum_{i=1}^{n} \hat{f}_i(x_i) + \sum_{1 \leq i < j \leq n} \hat{f}_{ij}(x_i, x_j)$.

end

4 An example of ARBF-HDMR

We employ a 3D analytical function

$$f(\mathbf{x}) = x_2^2 + x_1 x_3 + x_1 - 4, \quad \mathbf{x} \in [0, 1]^3 \tag{28}$$

to illustrate the ARBF-HDMR modeling process. This function has linear responses along variables x_1 and x_3, quadric responses along x_2, and the two variables x_1 and x_3 are correlated.

In the comparison, generally, we need to set a computational budget N_{max} for the studied problem. Thereafter, we run the RBF-HDMR and ARBF-HDMR modeling processes and evaluate their final prediction accuracies, respectively. But as stated before, the RBF-HDMR modeling process cannot stop with an arbitrary N_{max}; otherwise, it will produce an incomplete model. Hence, for the purpose of comparison, we first run the RBF-HDMR modeling process till convergence, and then use the number of points in RBF-HDMR as N_{max} for ARBF-HDMR.

The RBF-HDMR model for this function is constructed in a regular way following Liu et al. (2017b). The component RBF models are built with the multi-quadric basis function, and the related shape parameter s is calculated by an empirical formula in Hardy (1971). It is found that the RBF-HDMR requires 21 points to complete the modeling process.

Thus, to be fair, we set $N_{max} = 21$ and use the proposed approach to build the ARBF-HDMR models (ARBF_{MM}-HDMR, ARBF_{CVV}-HDMR and ARBF^b_{MM}-HDMR). The first fourteen points are used to build the initial RBF-HDMR model, and the remaining seven points are adaptively sampled on identified potential cuts. Note that for ARBF^b_{MM}-HDMR, the best-model strategy selects the best RBF model for each component function from three kinds of RBF models built with the multi-quadric, Gaussian and cubic basis functions, respectively.

Table 1 shows the modeling results of the RBF-HDMR and three ARBF-HDMR models for the test function. Here, two global criteria (R^2 and relative average absolute error (RAAE)) and a local criterion (relative maximum absolute error (RMAE)) are employed to assess the modeling performance. The details of these three criteria will be provided in the next section. Note that the best results, i.e., the largest R^2 value and the smallest RAAE and RMAE values, are marked in bold.

It is observed that, due to the adaptive modeling framework, all the three ARBF-HDMR models perform much better than the original RBF-HDMR model using the same number of points in terms of both global and local criteria. Among the three ARBF-HDMR models, ARBF^b_{MM}-HDMR provides much more accurate predictions, and ARBF_{CVV}-HDMR has a slightly better performance than ARBF_{MM}-HDMR.

Figure 4 depicts the component RBF models in the RBF-HDMR and three ARBF-HDMR modeling processes on three cut lines and the correlated cut plane for the 3D case. It is observed that the RBF-HDMR puts almost half of the points on the second cut line to build the RBF model $\hat{f}_2$. On the other hand, the ARBF-HDMR models identify that the correlated cut plane is also relevant and should be considered in order to efficiently improve the overall model accuracy. Thus, compared to the RBF-HDMR, the ARBF-HDMR models sample four more points on the cut plane, placing only six points on the second cut line. The test results reveal that, because of the local view, over-sampling (for $\hat{f}_2$) and under-sampling (for $\hat{f}_{13}$) occur in the RBF-HDMR, whereas the ARBF-HDMR models determine the

Table 1 The modeling results of RBF-HDMR and three ARBF-HDMR models for the 3D test case

Model	R^2	RAAE	RMAE	N_{max}
RBF-HDMR	9.9837E-01	3.4169E-02	8.2231E-02	21
ARBF_{MM}-HDMR	9.9965E-01	1.3972E-02	6.8501E-02	14+7
ARBF_{CVV}-HDMR	9.9968e-01	1.3299E-02	6.6558E-02	14+7
ARBF^{b}_{MM}-HDMR	**9.9996E − 01**	**4.9381E − 03**	**2.3151E − 02**	14+7

sample locations more reasonably from a global view, thus leading to more accurate predictions.

In addition, among the three ARBF-HDMR models, the ARBF_{CVV}-HDMR places points in a slightly different manner from that of the ARBF_{MM}-HDMR, which results in a slightly better performance. The ARBF^{b}_{MM}-HDMR and ARBF_{MM}-HDMR have the same sample distribution; however, the ARBF^{b}_{MM}-HDMR builds $\hat{f}_1$ and $\hat{f}_3$ using the multi-quadric basis function, while it builds $\hat{f}_2$ and $\hat{f}_{13}$ using the cubic basis function. It is observed that the best-model strategy significantly improves the prediction quality.

Furthermore, Fig. 5 shows the track of the 21 points in the RBF-HDMR and three ARBF-HDMR models, respectively. In this figure, the horizontal axis represents the point number and the vertical axis is the identity of the cut line (plane). For example, along the horizontal axis, the number "0" represents the center point of the domain, and the number "2" indicates the second point sampled in the modeling process; along the vertical axis, the number "2" means the second cut line with variable x_2, and the number "13" means the cut plane with variables x_1 and x_3.

It is observed that the RBF-HDMR samples the cuts independently and successively. It first samples two points in the first cut line because of the linearity, then nine points in the second cut line, then two points in the third cut line, and finally four points in the correlated cut plane. The rest of the points are the validation points to check for the existence of second-order terms and the correlation of two variables.

In the ARBF-HDMR modeling processes, the black circles represent the initial RBF-HDMR modeling process that captures the characteristics of the target function, whereas the red circles represent the subsequent adaptive modeling process. It is observed that after obtaining the initial RBF-HDMR model, the ARBF_{MM}-HDMR and ARBF_{CVV}-HDMR identify the potential of the second cut line and sample two points there. As $\hat{f}_2$ improves, the correlated cut plane becomes the potential one and is sampled with two points. Then, the two models switch back to the second cut line, and finally they switch again to the correlated cut plane. On the other hand, the ARBF^{b}_{MM}-HDMR first identifies the second cut line as the potential one and samples three points, and then captures the correlated cut plane to sample the rest of the four points. It is found that the ARBF-HDMRs determine the potential cuts in a *dynamic* way, which helps them outperform the RBF-HDMR. Here, "dynamic" means that, unlike the original RBF-HDMR that samples the cuts successively, the ARBF-HDMRs identify and sample a potential cut in each iteration via the error analysis described in Section 3.2.

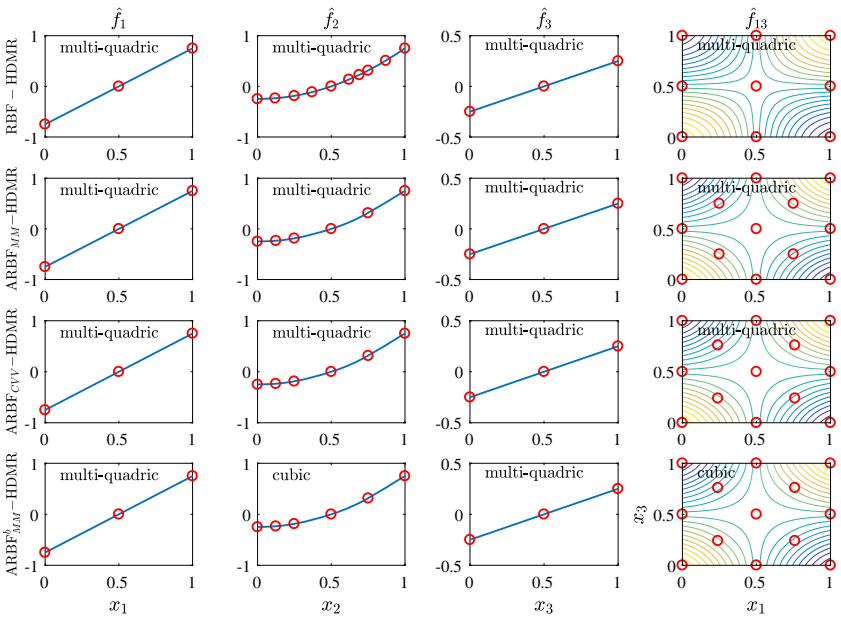

Fig. 4 The component RBF models in the RBF-HDMR and three ARBF-HDMR modeling processes on three cut lines and the correlated cut plane for the 3D test case

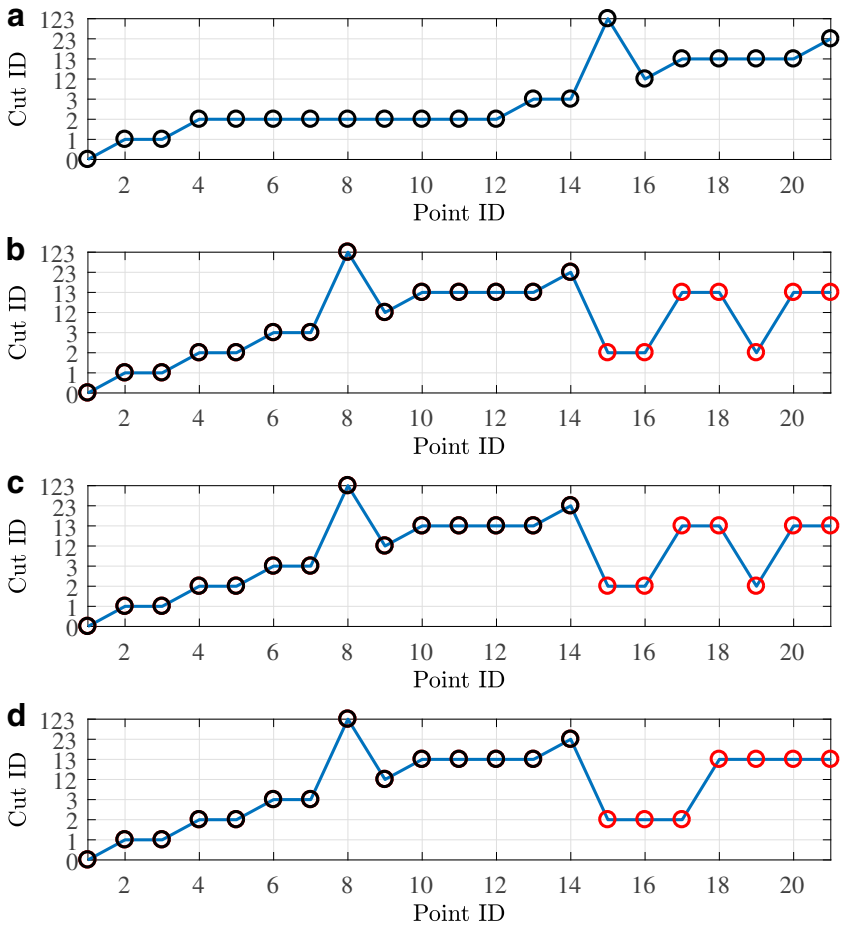

Fig. 5 The track of the 21 points sampled in the **a** RBF-HDMR, **b** ARBF$_{MM}$-HDMR, **c** ARBF$_{CVV}$-HDMR, and **d** ARBF$^{b}_{MM}$-HDMR modeling processes, respectively. In the three ARBF-HDMRs, the black circles represent the initial RBF-HDMR modeling process, whereas the red circles represent the subsequent adaptive modeling process

5 Additional numerical experiments

5.1 Test cases

To comprehensively assess the performance of the ARBF-HDMR, we employ five benchmark functions F1-F5 (Shan and Wang 2010a) and four engineering examples E1-E4, the expressions of which are provided in the Appendix.

The first 5D engineering example (E1) is the design of a direct methanol fuel cell system, in which methanol is used as the fuel to generate electricity via reaction with oxygen in the air (Yang and Xue 2015). We attempt to model the semi-empirical output voltage model of a specific fuel cell system, which is influenced by the current density I (A/cm^2), temperature T (K), methanol concentration C_{ME} (M), methanol flow rate F_{ME} (ccm), and air flow rate F_{AIR} (ccm).

The second 8D engineering example (E2) is a model that describes the flow of water through a borehole that is drilled from the ground surface through two aquifers (Morris et al. 1993). The design variables include the radius of the borehole r_w (m), radius of influence r (m), transmissivity of upper aquifer T_u (m^2/y), potentiometric head of upper aquifer H_u (m), transmissivity of lower aquifer T_l (m^2/y), potentiometric head of lower aquifer H_l (m), length of borehole L (m), and hydraulic conductivity of borehole K_w (m/y).

The third 10D engineering example (E3) is a conceptual level estimate of the weight of a light aircraft wing (Forrester et al. 2008). The design variables include the wing area S_w (ft^2), weight of fuel in the ring W_{fw} (lb), aspect ratio A, quarter-chord sweep Λ (deg), dynamic pressure at cruise q (lb/ft^2), taper ratio λ, aerofoil thickness to chord ratio tc, ultimate load factor N_z, flight design gross weight W_{dg} (lb), and paint weight W_p (lb/ft^2).

The last 30D engineering example (E4) attempts to model the tip deflection δ of a ten-stepped cantilever beam with a $P = 50$ kN force in the tip and a material of $E = 200$ GPa and $\sigma_{allow} = 350$ MPa (Cheng et al. 2015). The width b_i (m), height h_i (m) and length l_i (m) of each step are selected as design variables. This problem presents a high level of complexity as the global optimum is unknown.

5.2 Performance criteria

In the testing, two commonly used global performance criteria and a local performance criterion are employed to assess the performance of the ARBF-HDMR.

(1) R^2

$$R^2 = 1 - \frac{\sum_{i=1}^{t}\left[f(\mathbf{x}_i) - \hat{f}(\mathbf{x}_i)\right]^2}{\sum_{i=1}^{t}\left[f(\mathbf{x}_i) - \bar{f}\right]^2} \tag{29}$$

where $\mathbf{x}_i$ is one of the t validation points and $\bar{f}$ is the average response over the t points. The closer the value of R^2 approaches one, the more accurate the model is.

(2) RAAE

$$RAAE = \frac{\sum_{i=1}^{t} \mid f(\mathbf{x}_i) - \hat{f}(\mathbf{x}_i) \mid}{t \times \text{STD}} \tag{30}$$

where STD stands for standard deviation of the function responses at t validation points. An accurate model prefers a small RAAE value.

(3) RMAE

$$RMAE = \frac{\max_{1\le i\le t} \mid f(\mathbf{x}_i) - \hat{f}(\mathbf{x}_i) \mid}{\text{STD}} \tag{31}$$

This local metric measures the maximal prediction error in the design space. An accurate model prefers a small RMAE value.

In the following numerical experiments, the above three criteria are calculated using $t = 5000$ validation points generated by the Matlab routine *lhsdesign*.

5.3 Results and discussions

In the numerical experiments, the RBF-HDMR model is first built for each test case, and then the required number of points acts as the computational budget N_{max} for the ARBF-HDMR. Table 2 provides the modeling results of the RBF-HDMR and three ARBF-HDMR models for the five benchmark functions and four engineering examples. The best results are marked in bold.

5.3.1 ARBF-HDMR vs RBF-HDMR

It is observed that, in terms of global and local criteria, the ARBF-HDMR models perform much better than the RBF-HDMR for most of the test cases with the same number of points, especially for F2, F4, F5, E1 and E3.

The impressive performance of the ARBF-HDMRs directly comes from the adaptive and tractable modeling framework. As stated before, the RBF-HDMR independently and successively handles the component functions, which induces over-/under-sampling on some cuts. In contrast, the ARBF-HDMR determines which cut line or plane should be improved by considering them from a global perspective.

Figure 6 depicts the first-order component functions f_1 and f_2 for the E2 case. It is found that the function f_2 has a higher nonlinearity than f_1, but the responses of f_1 are several orders of magnitude larger than that of f_2. Thus, the accuracy of the RBF model $\hat{f}_1$ more significantly affects the overall model accuracy. However, because of the local view, the RBF-HDMR cannot identify the importance of f_1, and samples only eight points on this cut line while ten points on the second nonlinear cut line. In contrast, because of the global view, the ARBF-HDMR always focuses on improving the RBF models on the potential cuts. For instance, by comparing the GMSE errors of $\hat{f}_1$ and $\hat{f}_2$, the ARBF$_{MM}$-HDMR recognizes the importance of f_1 and samples thirteen points on this cut line while only three points for f_2. It is worth noting that in ARBF$_{MM}$-HDMR, the thirteen points on the first cut line are not sampled in successive iterations, but sampled when the cut line is identified as a potential one.

In addition to the local view, the over-/under-sampling problems in the RBF-HDMR are also caused by the fact that this approach uses a single test point to assess the accuracy of the component RBF model in order to determine whether the modeling should terminate or not. This point error estimation, however, is not a good representation of the RBF model accuracy. Oppositely, the ARBF-HDMR employs the more accurate and robust GMSE criterion to estimate the RBF model accuracy.

5.3.2 Comparison among three ARBF-HDMR models

All the three ARBF-HDMR models follow the adaptive and tractable modeling framework. They only differ in the employed sampling and modeling strategies to approximate the component functions on potential cuts.

The impact of different sampling strategies is first investigated by comparing the results of ARBF$_{MM}$-HDMR and ARBF$_{CVV}$-HDMR. The test results show that the ARBF$_{MM}$-HDMR outperforms the ARBF$_{CVV}$-HDMR in five out of the nine cases. Figure 7 shows the convergence curves of ARBF$_{MM}$-HDMR and ARBF$_{CVV}$-HDMR for the F2 case. It is observed that in the early stage, the two ARBF-HDMR models converge similarly; in the middle stage, the ARBF$_{CVV}$-HDMR converges faster; in the later stage, the addition of new points by the ARBF$_{CVV}$-HDMR, on the contrary, deteriorates the predictions.

We observed that for the five cases where ARBF$_{MM}$-HDMR outperforms ARBF$_{CVV}$-HDMR, ARBF$_{CVV}$-HDMR quickly reduces the $GMSE_I$ of $\mathcal{F}_I$ such that, after several iterations, $\mathcal{F}_{II}$ has an opportunity to be selected as the potential system because $GMSE_I < GMSE_{II}$; at this point, the true error of $\mathcal{F}_I$, however, is still larger than that of $\mathcal{F}_{II}$. The adaptive CVV sampling criterion is capable of efficiently decreasing the GMSE value of the RBF model,

Table 2 The modeling results of the RBF-HDMR and three ARBF-HDMR models for the five benchmark functions and four engineering examples

f	Model	R^2	RAAE	RMAE	N_{max}
F1	RBF-HDMR	7.0371E-01	4.4924E-01	1.7920E+00	297
	ARBF$_{MM}$-HDMR	7.9924E-01	3.5936E-01	1.5848E+00	202+95
	ARBF$_{CVV}$-HDMR	**9.6770E − 01**	**1.3836E − 01**	**1.0741E + 00**	202+95
	ARBF$^{b}_{MM}$-HDMR	8.5068E-01	3.1201E-01	1.3364E+00	202+95
F2	RBF-HDMR	9.9934E-01	2.4157E-02	5.6471E-02	347
	ARBF$_{MM}$-HDMR	**1.0000E + 00**	1.2316E-03	1.7318E-02	202+145
	ARBF$_{CVV}$-HDMR	9.9998E-01	3.3987E-03	1.9452E-02	202+145
	ARBF$^{b}_{MM}$-HDMR	**1.0000E + 00**	**6.5812E − 04**	**1.1853E − 02**	202+145
F3	RBF-HDMR	9.9027E-01	6.6834E-02	4.7703E-01	289
	ARBF$_{MM}$-HDMR	**9.9745E − 01**	3.3200E-02	**3.2207E − 01**	70+219
	ARBF$_{CVV}$-HDMR	9.9626E-01	3.7116E-02	5.0237E-01	70+219
	ARBF$^{b}_{MM}$-HDMR	9.9730E-01	**3.3065E − 02**	3.4684E-01	70+219
F4	RBF-HDMR	9.9828E-01	3.2779E-02	1.4543E-01	109
	ARBF$_{MM}$-HDMR	9.9932E-01	2.1112E-02	9.5776E-02	70+39
	ARBF$_{CVV}$-HDMR	9.9963E-01	1.5777E-02	5.7993E-02	70+39
	ARBF$^{b}_{MM}$-HDMR	**9.9994E − 01**	**6.1562E − 03**	**2.7543E − 02**	70+39
F5	RBF-HDMR	8.9448E-01	2.5644E-01	1.5709E+00	320
	ARBF$_{MM}$-HDMR	9.9101E-01	7.4883E-02	4.1593E-01	244+76
	ARBF$_{CVV}$-HDMR	9.9730E-01	4.0927E-02	2.3815E-01	244+76
	ARBF$^{b}_{MM}$-HDMR	**9.9879E − 01**	**2.7317E − 02**	**1.5761E − 01**	244+76
E1	RBF-HDMR	9.8486E-01	5.3289E-02	7.4436E-01	223
	ARBF$_{MM}$-HDMR	9.9838E-01	2.2768E-02	3.0573E-01	43+180
	ARBF$_{CVV}$-HDMR	9.9883E-01	1.8442E-02	2.7379E-01	43+180
	ARBF$^{b}_{MM}$-HDMR	**9.9937E − 01**	**1.6314E − 02**	**2.3890E − 01**	43+180
E2	RBF-HDMR	9.9923E-01	1.8109E-02	1.6513E-01	211
	ARBF$_{MM}$-HDMR	9.9949E-01	1.4420E-02	1.6297E-01	73+138
	ARBF$_{CVV}$-HDMR	9.9950E-01	1.5682E-02	1.4985E-01	73+138
	ARBF$^{b}_{MM}$-HDMR	**9.9958E − 01**	**1.2755E − 02**	**1.4380E − 01**	73+138
E3	RBF-HDMR	9.6251E-01	1.2345E-01	**1.2793E + 00**	488
	ARBF$_{MM}$-HDMR	9.7695E-01	8.1148E-02	1.4351E+00	100+388
	ARBF$_{CVV}$-HDMR	9.7175E-01	1.0123E-01	1.3733E+00	100+388
	ARBF$^{b}_{MM}$-HDMR	**9.7769E − 01**	**7.9365E − 02**	1.4285E+00	100+388
E4	RBF-HDMR	9.5039E-01	1.6071E-01	1.6545E+00	939
	ARBF$_{MM}$-HDMR	9.5924E-01	1.4646E-01	**1.6198E + 00**	599+340
	ARBF$_{CVV}$-HDMR	9.3484E-01	1.8670E-01	1.6906E+00	599+340
	ARBF$^{b}_{MM}$-HDMR	**9.6371E − 01**	**1.3315E − 01**	1.6315E+00	599+340

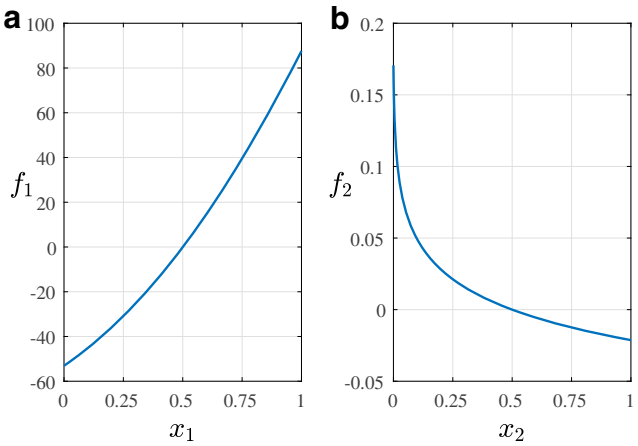

Fig. 6 The plots of **a** f_1 and **b** f_2 for the E2 case

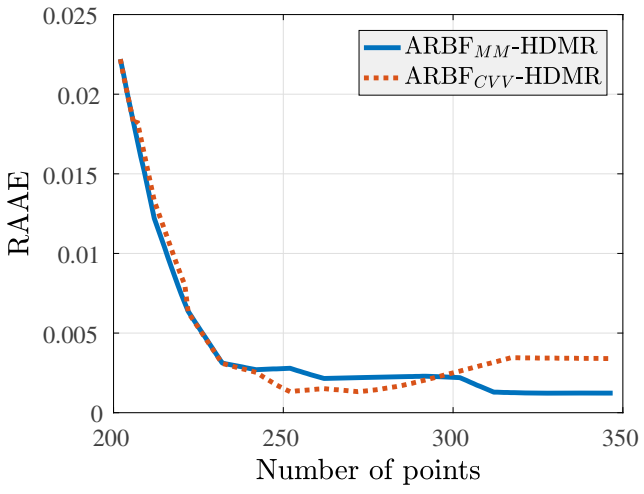

Fig. 7 The convergence curves of ARBF$_{MM}$-HDMR and ARBF$_{CVV}$-HDMR for the F2 case

since it sequentially adds new points in regions with the largest LOO error via (25). However, the true model error may not so quickly decrease. As a result, the discrepancy may lead to the wrong identification of the potential system. For example, for the F2 case in Fig. 7, the potential cuts identified by ARBF$_{MM}$-HDMR and ARBF$^{b}_{MM}$-HDMR are cut lines in all the iterations; whereas ARBF$_{CVV}$-HDMR begins to identify cut planes as potential cuts after 272 simulations and, correspondingly, the convergence curve starts to increase. Therefore, the space-filling maximin sampling criterion is recommended for ARBF-HDMR.

The impact of different modeling strategies is next investigated by comparing the results of ARBF$_{MM}$-HDMR and ARBF$^{b}_{MM}$-HDMR. It is observed that compared to the fixed RBF model used in ARBF$_{MM}$-HDMR, the best-model strategy significantly improves the prediction quality of ARBF$^{b}_{MM}$-HDMR in all cases in terms of RAAE and in seven cases in terms of RMAE. Taking the E1 case for example, the initial RBF-HDMR identifies that five first-order component functions and a second-order component function should be approximated. As a result, the final ARBF$_{MM}$-HDMR model uses the fixed multi-quadric basis function to build all the six component RBF models; in contrast, the final ARBF$^{b}_{MM}$-HDMR model uses the cubic basis function to build four component RBF models and the Gaussian basis function to build the remaining two RBF models, which leads to more accurate predictions.

5.4 Other discussions

Taking the ARBF$^{b}_{MM}$-HDMR as the best version of ARBF-HDMR, this section discusses the impact of model order and the running time in ARBF-HDMR implementations.

5.4.1 Impact of model order

Table 3 shows the modeling results of the first-order RBF-HDMR and ARBF$^{b}_{MM}$-HDMR for the nine test cases. Note that due to the ignorance of the second-order terms, the N_{ini} value for the first-order ARBF$^{b}_{MM}$-HDMR is fixed at $1+2n$, where "1" denotes the cut point and "$2n$" denotes the points on n cut lines.

It is observed that the first-order ARBF$^{b}_{MM}$-HDMR still outperforms the first-order RBF-HDMR. In terms of the two global criteria, the ARBF$^{b}_{MM}$-HDMR performs better for all the nine cases. In terms of the local criterion, the ARBF$^{b}_{MM}$-HDMR performs better in seven out of the nine cases.

By comparing the results in Tables 2 and 3, it is observed that the first-order RBF-HDMR model outperforms the second-order version for F2 and E3. Shan and Wang (2010a) attributed this behavior to the fact that the errors in RBF construction may be larger than the impact of higher-order terms, and thus leads to the over-fitting of RBF models. We believe that it is due to the passive RBF-HDMR modeling framework that cannot identify the cuts to be focused on. Oppositely, due to the global view, the second-order ARBF$^{b}_{MM}$-HDMR model always outperforms the first-order model.

It may be argued that the computational budget N_{max} is not the same for the first- and second-order ARBF$^{b}_{MM}$-HDMR models in Tables 2 and 3. For this reason, we take F2 and E3 for example, and run the first-order ARBF$^{b}_{MM}$-HDMR modeling process until the number of points is the same as that for the second-order model. Figure 8 depicts the convergence histories of the first- and second-order ARBF$^{b}_{MM}$-HDMR models for F2 and E3, respectively. It is

Table 3 The modeling results of the first-order RBF-HDMR and ARBF$^{b}_{MM}$-HDMR for the five benchmark functions and four engineering examples

f	Model	R^2	RAAE	RMAE	N_{max}
F1	RBF-HDMR	6.3053E-01	4.8151E-01	3.4103E+00	91
	ARBF$^{b}_{MM}$-HDMR	**7.3624E − 01**	**3.9712E − 01**	**3.1155E + 00**	21+70
F2	RBF-HDMR	9.9930E-01	2.3865E-02	1.0429E-01	31
	ARBF$^{b}_{MM}$-HDMR	**9.9991E − 01**	**6.8295E − 03**	**7.6987E − 02**	21+10
F3	RBF-HDMR	9.9023E-01	6.6980E-02	4.7617E-01	181
	ARBF$^{b}_{MM}$-HDMR	**9.9692E − 01**	**3.7582E − 02**	**3.0727E − 01**	21+160
F4	RBF-HDMR	9.9705E-01	4.2718E-02	2.0483E-01	53
	ARBF$^{b}_{MM}$-HDMR	**9.9860E − 01**	**2.8303E − 02**	**1.2525E − 01**	21+32
F5	RBF-HDMR	8.4039E-01	3.5858E-01	1.1374E+00	49
	ARBF$^{b}_{MM}$-HDMR	**9.4318E − 01**	**2.0209E − 01**	**9.6321E − 01**	33+16
E1	RBF-HDMR	9.8422E-01	8.0646E-02	7.3105E-01	34
	ARBF$^{b}_{MM}$-HDMR	**9.9024E − 01**	**6.6024E − 02**	**7.1869E − 01**	11+23
E2	RBF-HDMR	**9.6144E − 01**	1.3724E-01	1.1557E+00	47
	ARBF$^{b}_{MM}$-HDMR	9.6142E-01	**1.3690E − 01**	**1.1545E + 00**	17+30
E3	RBF-HDMR	9.4443E-01	1.1423E-01	**1.7548E + 00**	90
	ARBF$^{b}_{MM}$-HDMR	**9.4945E − 01**	**1.0789E − 01**	1.8463E+00	21+69
E4	RBF-HDMR	7.8764E-01	3.0175E-01	**3.6137E + 00**	171
	ARBF$^{b}_{MM}$-HDMR	**7.8925E − 01**	**2.9988E − 01**	3.6788E+00	61+110

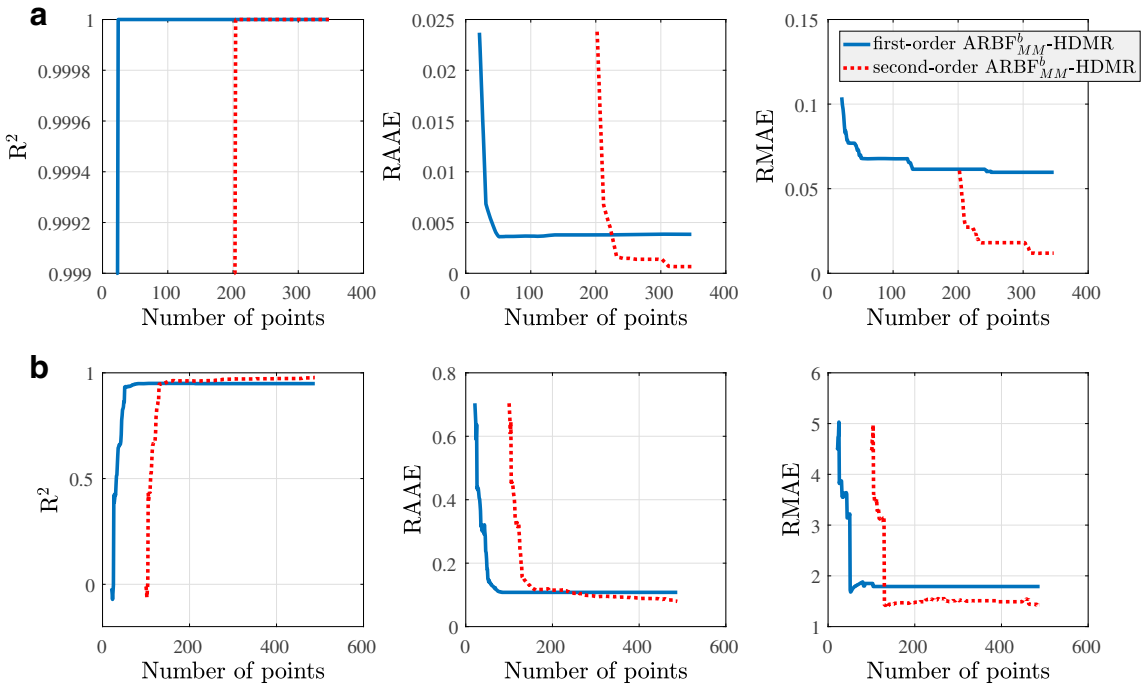

Fig. 8 The convergence histories of the first-order and second-order ARBF$^b_{MM}$-HDMR models for **a** F2 and **b** E3, respectively

found that in the early stage, the second-order ARBF$^b_{MM}$-HDMR needs to spend more points on the second-order terms, whereas the first-order ARBF$^b_{MM}$-HDMR ignores them and focuses all its attention on the improvement of the first-order RBF models. Hence, the first-order model converges faster in this stage. Thereafter, as the modeling process evolves, the first-order ARBF$^b_{MM}$-HDMR is incapable to further improve the model accuracy because of the loss of second-order terms. The comparison results reveal that the consideration of second-order terms enlarges the space of component functions such that the ARBF$^b_{MM}$-HDMR can determine the locations of points more effectively.

The above discussions offer the guidelines for the practical use of ARBF$^b_{MM}$-HDMR: the second-order version is recommended if N_{max} is large; on the other hand, the first-order version becomes a better choice if N_{max} is strictly limited, e.g., $2n + 1 < N_{max} \ll 2n^2 + 2$.

5.4.2 Running time

Table 4 provides the running times of RBF-HDMR and ARBF$^b_{MM}$-HDMR for the nine cases to illustrate the modeling efficiency. All the numerical experiments are executed in a Matlab environment with an Intel 3.40 GHz processor. It is found that the ARBF$^b_{MM}$-HDMR requires much more computing time than the RBF-HDMR, particularly for cases with a large sample size and high dimensionality (e.g., E3 and E4).

The running time of ARBF$^b_{MM}$-HDMR mainly comes from the cross-validation process of each RBF model. Assume that any two input variables of the target function are correlated and each RBF model has m sampled points, then we should refit the RBF models $mn(n + 1)/2$ times in order to estimate the GMSE values. This becomes more and more time-consuming with the increase of sample size and dimensionality. To alleviate the computing cost, the GMSE can be alternatively computed in a parallel fashion.

It is worth noting that the ARBF$^b_{MM}$-HDMR model is proposed for expensive simulation-based problems, in which a simulation may require several hours or even days (Mueller et al. 2013). Thus, in practice, the running time of ARBF$^b_{MM}$-HDMR is negligible compared to the time spent on simulations. Besides, though the running time of ARBF$^b_{MM}$-HDMR itself is much larger than that of RBF-HDMR, the ARBF$^b_{MM}$-HDMR makes a great saving

Table 4 The running times (s) of RBF-HDMR and ARBF$^b_{MM}$-HDMR for the five benchmark functions and four engineering examples

Model	F1	F2	F3	F4	F5	E1	E2	E3	E4
RBF-HDMR	1.4	0.4	1.6	0.2	0.4	0.4	0.3	5.1	1.9
ARBF$^b_{MM}$-HDMR	120.2	184.6	70.5	11.3	78.8	91.6	66.6	531.7	604.2

in the time spent on simulations. For example, the RBF-HDMR requires 939 points (simulations) to achieve the model accuracy of RAAE = 0.1607 for the 30D E4 case, whereas the ARBF$^{b}_{MM}$-HDMR only needs 694 points to achieve the same model accuracy.

6 Concluding remarks

This article proposed an adaptive and tractable RBF-HDMR modeling approach for high-dimensional problems under limited computational budget. The extensive numerical results offer the following findings:

- Compared to the original RBF-HDMR, the ARBF-HDMR provides more accurate predictions in terms of both the global and local criteria with the same number of points;
- Among the different versions of ARBF-HDMR, the version using the maximin sampling criterion and the best-model strategy is recommended;
- the second-order ARBF$^{b}_{MM}$-HDMR always outperforms the first-order version due to the enlarged space of component functions. However, if the computational budget is strictly limited, e.g., $2n + 1 < N_{max} \ll 2n^2 + 2$, the first-order model is better.

For a black-box target function, the main disadvantage of the proposed ARBF-HDMR is that the N_{ini} value in the second-order modeling process cannot be determined in advance, since it depends on both the characteristics and the dimensionality of the target function.

In this article, the ARBF-HDMR uses the RBF model to approximate the component functions. It is worth noting that the proposed approach can use some other metamodel types (e.g., Kriging and SVR), and even effective ensemble modeling strategies (Liu et al. 2016c; Goel et al. 2007; Acar and Rais-Rohani 2009) for approximation purposes.

Acknowledgements The majority of this work was finished before joining the Lab. We appreciate the support from the National Research Foundation (NRF) Singapore under the Corp Lab@University Scheme for completing the research. It is also partially supported by the Data Science and Artificial Intelligence Research Center (DSAIR) and the School of Computer Science and Engineering at Nanyang Technological University.

Appendix

Tables 5 and 6 offer the expressions of the employed five benchmark functions and four engineering examples, respectively.

Table 5 Five benchmark functions

ID	Expression	Range
F1	$f(\mathbf{x}) = \sum_{i=1}^{10} \left[(\ln(x_i - 2))^2 + (\ln(10 - x_i))^2\right] - \left(\prod_{i=1}^{10} x_i\right)^{0.2}$	$x_i \in [2.1, 9.9]$
F2	$f(\mathbf{x}) = \sum_{i=1}^{10} x_i \left(c_i + \ln \frac{x_i}{x_1 + \cdots + x_{10}}\right)$	$x_i \in [1e^{-6}, 10]$
F3	$f(\mathbf{x}) = \sum_{i=1}^{10} e^{x_i} \left[c_i + x_i - \ln\left(\sum_{k=1}^{10} e^{x_k}\right)\right]$	$x_i \in [-10, 10]$
F4	$f(\mathbf{x}) = x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2$ $+2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45$	$x_i \in [-10, 11]$
F5	$f(\mathbf{x}) = \sum_{i=1}^{16} \sum_{j=1}^{16} a_{ij}(x_i^2 + x_i + 1)(x_j^2 + x_j + 1)$	$x_i \in [0, 5]$

For F2 and F3:

$$c_{1 \le i \le 10} = -6.089, -17.164, -34.054, -5.914, -24.721, -14.986, -24.1, -10.708, -26.662, -22.179.$$

For F5:

$$[a_{ij}]_{rows1-8} = \begin{bmatrix} 1&0&0&1&0&0&1&1&0&0&0&0&0&0&0&1 \\ 0&1&1&0&0&0&1&0&0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&1&0&1&1&0&0&0&1&0&0 \\ 0&0&0&1&0&0&1&0&0&0&1&0&0&0&1&0 \\ 0&0&0&0&1&1&0&0&0&1&0&1&0&0&0&1 \\ 0&0&0&0&0&1&0&1&0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&1&0&0&0&1&0&1&0&0&0 \\ 0&0&0&0&0&0&0&1&0&1&0&0&0&0&1&0 \end{bmatrix}$$

$$[a_{ij}]_{rows9-16} = \begin{bmatrix} 0&0&0&0&0&0&0&0&1&0&0&1&0&0&0&1 \\ 0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&1&0&1&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&1&0&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \end{bmatrix}$$

Table 6 Four engineering examples

ID	Expression	Range
E1	$f_V = 1.21 - 3.7534 \times 10^{-5} T - 3.1534 \times 10^{-4} T \ln C_{ME}$	
	$+6.62 \times 10^{-5} T \ln F_{AIR} - 0.7499 - 6.9897 e^{\left(\frac{916.91}{T} - 4.6392\right)} I$	
	$-[1.2658 \times 10^5 I^3 + 46196 I^2 - 4281 I - 0.4029 T$	
	$-18.8094 C_{ME}^2 + 18.8094 C_{ME} + 10.496]$	$I \in [0.0003, 0.08]$, $T \in [298, 343]$
	$\times[\ln I - 3.9056 + 2.9582$	$C_{ME} \in [0.25, 2]$, $F_{ME} \in [3.5, 5.5]$
	$\times 10^{-4} \left(\ln C_{ME} + \ln\left(1 - \frac{1}{5.3466 \times 10^7 e^{(-5182.4/T)} C_{ME}^2} I\right)\right)]$	$F_{AIR} \in [81.2, 140.8]$
	$-[-1.2687 \times 10^5 I^3 - 46221 I^2 + 4283.6 I + 0.4033 T$	
	$+18.818 C_{ME}^2 - 18.818 C_{ME} - 10.572]$	
	$\times[\ln I - 3.8959 - 8.2402 \times 10^{-4} \ln F_{AIR}] + 31.583 I^2 \ln F_{ME}$	
E2		$r_w \in [0.05, 0.15]$, $r \in [100, 50000]$
	$f_{flow} = \frac{2\pi T_u (H_u - H_l)}{\ln \frac{r}{r_w}\left[1 + \frac{2LT_u}{\ln \frac{r}{r_w} r_w^2 K_w} + \frac{T_u}{T_l}\right]}$	$T_u \in [63070, 115600]$, $H_u \in [990, 1110]$
		$T_l \in [63.1, 116]$, $H_l \in [700, 820]$
		$L \in [1120, 1680]$, $K_2 \in [9855, 12045]$
E3		$S_w \in [150, 200]$, $W_{fw} \in [220, 300]$
		$A \in [6, 10]$, $\Lambda \in [-10, 10]$
	$f_w = 0.036 S_w^{0.758} W_{fw}^{0.0035} \left(\frac{A}{\cos^2 \Lambda}\right) q^{0.006} \lambda^{0.04}$	$q \in [16, 15]$, $\lambda \in [0.5, 1]$
	$\times \left(\frac{100 tc}{\cos \Lambda}\right)^{-0.3} (N_z W_{dg})^{0.49} + S_w W_p$	$tc \in [0.08, 0.18]$, $N_z \in [2.5, 6]$
		$W_{dg} \in [1700, 2500]$, $W_p \in [0.025, 0.08]$
E4	$f_\delta = \int_0^{l_d} \frac{P x_d^2}{E I_d} dx_d + \int_0^{l_{d-1}} \frac{P(x_{d-1} + l_d)^2}{E I_{d-1}} dx_{d-1} + \cdots$	$b_i \in [0.01, 0.05]$
	$+ \int_0^{l_1} \frac{P(x_1 + l_2 + l_3 + \cdots + l_d)^2}{E I_1} dx_1$	$h_i \in [0.3, 0.65]$
	$= \frac{P}{3E} \sum_{i=1}^{d} \left[\frac{12}{b_i h_i^3}\left(\left(\sum_{j=i}^{d} l_j\right)^3 - \left(\sum_{j=i+1}^{d} l_j\right)^3\right)\right]$	$l_i \in [0.5, 1]$, $1 \le i \le 10$

References

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

Andrews DW, Whang YJ (1990) Additive interactive regression models: circumvention of the curse of dimensionality. Econometric Theory 6(4):466–479

Breiman L, Friedman J, Stone CJ, Olshen RA (1984) Classification and regression trees. CRC press, Boca Raton

Cai X, Qiu H, Gao L, Yang P, Shao X (2016) An enhanced RBF-HDMR integrated with an adaptive sampling method for approximating high dimensional problems in engineering design. Struct Multidiscip Optim 53(6):1209–1229

Cheng GH, Younis A, Hajikolaei KH, Wang GG (2015) Trust region based mode pursuing sampling method for global optimization of high dimensional design problems. J Mech Des 137(2):021–407

Chowdhury R, Rao B (2009) Hybrid high dimensional model representation for reliability analysis. Comput Methods Appl Mech Eng 198(5):753–765

Crombecq K, Gorissen D, Deschrijver D, Dhaene T (2011a) A novel hybrid sequential design strategy for global surrogate modeling of computer experiments. SIAM J Sci Comput 33(4):1948–1974

Crombecq K, Laermans E, Dhaene T (2011b) Efficient space-filling and non-collapsing sequential design strategies for simulation-based modeling. Eur J Oper Res 214(3):683–696

Fang H, Horstemeyer MF (2006) Global response approximation with radial basis functions. Eng Optim 38(4):407–424

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

Friedman JH, Stuetzle W (1981) Projection pursuit regression. J Am Stat Assoc 76(376):817–823

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

Hardy RL (1971) Multiquadric equations of topography and other irregular surfaces. J Geophys Res 76(8):1905–1915

Huang Z, Qiu H, Zhao M, Cai X, Gao L (2015) An adaptive SVR-HDMR model for approximating high dimensional problems. Eng Comput 32(3):643–667

Johnson ME, Moore LM, Ylvisaker D (1990) Minimax and maximin distance designs. J Stat Plan Inference 26(2):131–148

Li E, Wang H, Li G (2012) High dimensional model representation (HDMR) coupled intelligent sampling strategy for nonlinear problems. Comput Phys Commun 183(9):1947–1955

Li G, Rosenthal C, Rabitz H (2001a) High dimensional model representations. J Phys Chem A 105(33):7765–7777

Li G, Wang SW, Rosenthal C, Rabitz H (2001b) High dimensional model representations generated from low dimensional data samples. i. mp-Cut-HDMR. J Math Chem 30(1):1–30

Li G, Wang SW, Rabitz H (2002) Practical approaches to construct RS-HDMR component functions. J Phys Chem A 106(37):8721–8733

Li G, Hu J, Wang SW, Georgopoulos PG, Schoendorf J, Rabitz H (2006) Random sampling-high dimensional model representation (RS-HDMR) and orthogonality of its different order component functions. J Phys Chem A 110(7):2474–2485

Li G, Rabitz H, Hu J, Chen Z, Ju Y (2008) Regularized random-sampling high dimensional model representation (RS-HDMR). J Math Chem 43(3):1207–1232

Liu H, Xu S, Wang X (2015) Sequential sampling designs based on space reduction. Eng Optim 47(7):867–884

Liu H, Xu S, Ma Y, Chen X, Wang X (2016a) An adaptive bayesian sequential sampling approach for global metamodeling. J Mech Des 138(1):011–404

Liu H, Xu S, Wang X (2016b) Sampling strategies and metamodeling techniques for engineering design: comparison and application. In: ASME Turbo Expo 2016: Turbomachinery Technical Conference and Exposition, ASME, pp V02CT45A019–V02CT45A019

Liu H, Xu S, Wang X, Meng J, Yang S (2016c) Optimal weighted pointwise ensemble of radial basis functions with different basis functions. AIAA J 54(10):3117–3133

Liu H, Ong YS, Cai J (2017a) An adaptive sampling approach for kriging metamodeling by maximizing expected prediction error . Comput Chem Eng 106:171–182

Liu H, Wang X, Xu S (2017b) Generalized radial basis function-based high-dimensional model representation handling existing random data. J Mech Des 139(1):011–404

Liu Y, Hussaini MY, Ökten G (2016d) Accurate construction of high dimensional model representation with applications to uncertainty quantification. Reliab Eng Syst Saf 152:281–295

Morris MD, Mitchell TJ, Ylvisaker D (1993) Bayesian design and analysis of computer experiments: use of derivatives in surface prediction. Technometrics 35(3):243–255

Mueller L, Alsalihi Z, Verstraete T (2013) Multidisciplinary optimization of a turbocharger radial turbine. J Turbomach 135(2):021–022

Rabitz H, Aliş ÖF (1999) General foundations of high-dimensional model representations. J Math Chem 25(2):197–233

Rabitz H, Aliş ÖF, Shorter J, Shim K (1999) Efficient input-output model representations. Comput Phys Commun 117(1-2):11–20

Razavi S, Tolson BA, Burn DH (2012) Review of surrogate modeling in water resources. Water Resour Res 48(7):1–32

Rippa S (1999) An algorithm for selecting a good value for the parameter c in radial basis function interpolation. Adv Comput Math 11(2):193–210

Shan S, Wang GG (2010a) Metamodeling for high dimensional simulation-based design problems. J Mech Des 132(5):051–009

Shan S, Wang GG (2010b) Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions. Struct Multidiscip Optim 41(2):219–241

Shan S, Wang GG (2011) Turning black-box functions into white functions. J Mech Des 133(3):031–003

Sobol IM (1993) Sensitivity estimates for nonlinear mathematical models. Math Model Comput Exper 1(4):407–414

Sobol IM (2003) Theorems and examples on high dimensional model representation. Reliab Eng Syst Saf 79(2):187–193

Tang L, Wang H, Li G (2013) Advanced high strength steel springback optimization by projection-based heuristic global search algorithm. Mater Des 43:426–437

Tunga MA, Demiralp M (2005) A factorized high dimensional model representation on the nodes of a finite hyperprismatic regular grid. Appl Math Comput 164(3):865–883

Ulaganathan S, Couckuyt I, Dhaene T, Degroote J, Laermans E (2016) High dimensional kriging metamodelling utilising gradient information. Appl Math Model 40(9):5256–5270

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

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

Wang SW, Georgopoulos PG, Li G, Rabitz H (2003) Random sampling- high dimensional model representation (RS-HDMR) with nonuniformly distributed variables: Application to an integrated multimedia/multipathway exposure and dose model for trichloroethylene. J Phys Chem A 107(23):4707–4716

Xu S, Liu H, Wang X, Jiang X (2014) A robust error-pursuing sequential sampling approach for global metamodeling based on voronoi diagram and cross validation. J Mech Des 136(7):071–009

Yang Q, Xue D (2015) Comparative study on influencing factors in adaptive metamodeling. Eng Comput 31(3):561–577