A Robust Error-Pursuing Sequential Sampling Approach for Global Metamodeling Based on Voronoi Diagram and Cross Validation

Shengli Xu
School of Energy and Power Engineering,
Dalian University of Technology,
Dalian 116024, China
e-mail: xusl@dlut.edu.cn

Haitao Liu
School of Energy and Power Engineering,
Dalian University of Technology,
Dalian 116024, China
e-mail: lht@mail.dlut.edu.cn

Xiaofang Wang[1]
School of Energy and Power Engineering,
Dalian University of Technology,
Dalian 116024, China
e-mail: dlwxf@dlut.edu.cn

Xiaomo Jiang
School of Energy and Power Engineering,
Dalian University of Technology,
Dalian 116024, China
e-mail: jiang.xiaomo@gmail.com

Surrogate models are widely used in simulation-based engineering design and optimization to save the computing cost. The choice of sampling approach has a great impact on the metamodel accuracy. This article presents a robust error-pursuing sequential sampling approach called cross-validation (CV)-Voronoi for global metamodeling. During the sampling process, CV-Voronoi uses Voronoi diagram to partition the design space into a set of Voronoi cells according to existing points. The error behavior of each cell is estimated by leave-one-out (LOO) cross-validation approach. Large prediction error indicates that the constructed metamodel in this Voronoi cell has not been fitted well and, thus, new points should be sampled in this cell. In order to rapidly improve the metamodel accuracy, the proposed approach samples a Voronoi cell with the largest error value, which is marked as a sensitive region. The sampling approach exploits locally by the identification of sensitive region and explores globally with the shift of sensitive region. Comparative results with several sequential sampling approaches have demonstrated that the proposed approach is simple, robust, and achieves the desired metamodel accuracy with fewer samples, that is needed in simulation-based engineering design problems. [DOI: 10.1115/1.4027161]

Introduction

Surrogate models (also known as metamodels) have been widely used to approximate the real models in order to facilitate the engineering design and optimization. There are a number of commonly used metamodels, such as polynomial functions, Kriging models [1], radial basis functions [2,3], and multivariate adaptive regression splines [4]. Refer to Wang and Shan [5] for a detailed overview on various metamodeling techniques.

In the surrogate modeling of an engineering problem, extensive simulations are needed at sampled points in order to construct a metamodel with acceptable prediction accuracy. A challenging issue is how to achieve an accurate metamodel with samples as few as possible. Therefore, the sampling approaches (also known as design of computer experiments) play an important role in determining the metamodel accuracy.

In general, conventional sampling approaches can be classified into two categories: one-stage sampling approaches and sequential sampling approaches. A wide variety of traditional one-stage sampling approaches are available, such as Latin hypercube design (LHD) [6] and orthogonal array [7]. In recent years, numerous researchers have developed more effective and efficient one-stage sampling approaches, such as the columnwise-pairwise algorithm [8], the iterated local search heuristics algorithm [9], the translational propagation algorithm (TPLHD) [10], and the successive local enumeration algorithm [11]. There is a common issue among these one-stage sampling approaches, that is, the number of samples must be determined in advance. In practice, little information about the real model can be used to predetermine the proper sample size.

To address the shortcomings of one-stage sampling strategies, sequential sampling (also known as adaptive sampling or active learning) strategies are proposed to improve the metamodel accuracy over the entire design space of interest by sequentially selecting samples. In terms of the ultimate goal, sequential sampling approaches can be used for metamodel-based design optimization or global metamodeling. The former sequential sampling approaches focus on obtaining the global optimum by a trade-off between a search for the optimum of a metamodel and a search for unexplored regions to avoid missing promising areas due to inaccuracy of the metamodel [12], such as efficient global optimization [13], surrogate management framework [14], sequential design optimization [15], and model-assisted grid search [16]. The latter sequential approaches focus on sequentially improving accuracy of the metamodel over the entire design space. Maximin criterion has been commonly used in these approaches to use the coordinate information of existing points in the design space. Such sequential sampling approaches include minimax and maximin designs [17], Audze-Eglais design [18], centered L_2 discrepancy [19,20], MSE approach [12], etc. Sequential sampling approaches [21,22] have, also, been developed to combine the generally used maximin criterion with LHD [6,20] to obtain samples with both good space-filling and projective properties. For instance, Xiong et al. [21] proposed a sequential quasi-LHD sampling approach to overcome the drawback of maximin designs in high dimensions. Crombecq et al. [22] presented a sequential approach called mc-intersite-proj-th (MIPT) by using maximin criterion and a threshold function to obtain the space-filling and projective properties, respectively. Furthermore, some sequential sampling approaches have been developed to use not only the information of existing points but, also, the previous metamodels to locate new samples. These methods have been demonstrated to

[1]Corresponding author.

Contributed by the Design Automation Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received August 2, 2013; final manuscript received February 23, 2014; published online April 28, 2014. Assoc. Editor: Xiaoping Du.

outperform the aforementioned approaches in terms of the metamodel accuracy [15,23–25].

For a complex function, intuitively, one would expect to sample more points in the highly nonlinear regions and fewer points in the relatively flat regions in order to capture the behavior of this function with a smaller sample size. Therefore, the trade-off between global exploration and local exploitation plays a key role on the performance of sequential sampling approaches [26,24,27]. Turner et al. [28] employed a simulated-annealing approach to balance the local exploration and global exploitation explicitly. Li and Aute [25,29] proposed the cross-validation-based sequential sampling approaches (ACcumulative Error, ACE; Space-Filling Cross-Validation Tradeoff, SFCVT) to address this issue. Their methods utilize the prediction errors of samples obtained by LOO cross-validation method to sample more points on highly nonlinear regions and relatively sparse points on flat regions. This strategy uses the error information of current samples to construct an error-metamodel to probe errors of unexplored regions. A distance constraint is employed to prevent new points from clustering around existing points. Crombecq et al. [30] proposed a novel hybrid sequential strategy (LOLA-Voronoi) by using the local linear approximation to recognize the nonlinearity of regions and Voronoi tessellation to explore the whole design space. This sampling approach simply gives fixed weights of exploration and exploitation, which may lead to bad performance in some circumstances.

In this paper, we present a robust error-pursuing sequential sampling approach for global metamodeling in simulation-based engineering design and optimization. In each iteration, the entire design space is partitioned into a set of Voronoi cells according to the current samples by Voronoi diagram. A Voronoi cell is a surrounding region of an existing point. Its error behavior can be simply represented by the prediction error of this existing point. Large prediction error indicates that the constructed metamodel in this Voronoi cell has not been fitted well and, thus, new points should be sampled in this cell. In the proposed approach, a Voronoi cell with the largest error is marked as the sensitive region and sampled by using the maximin criterion. By always focusing on sampling the region with the largest error, the proposed approach is able to improve the metamodel accuracy rapidly. In addition, due to the partition of the design space and the identification of the sensitive region, the proposed algorithm pays attention to the local region and is able to avoid the additional optimization required by the methods presented in Li and Aute [25,29], thus, reducing the complexity of sampling process.

CV-Voronoi Approach

The proposed CV-Voronoi approach intends to improve the metamodel accuracy by sampling a region with the largest prediction error. If the existing point in this region is removed, the predicted response of the metamodel constructed by the rest of existing points will be far away from the actual response. Therefore, this region is referred to as a sensitive region and should be sampled more points in order to improve the metamodel accuracy.

Three major issues of the proposed sequential sampling approach need to be handled: (1) how to properly partition the design space according to existing samples, (2) how to identify a sensitive region, and (3) how to determine the location of new sample in the obtained sensitive region. The remainder subsections will present detailed descriptions to address these issues.

Partition the Design Space. The first step of the proposed methodology is to partition the entire design space in order to help designers to concentrate on local regions rather than the whole design space in the following steps. Assume a set of points $P = \{p_1, p_2, \ldots, p_m\} \in \mathbb{R}^d$ represent the current samples. In this study, Voronoi diagram approach is employed to partition the entire design space into a set of Voronoi cells

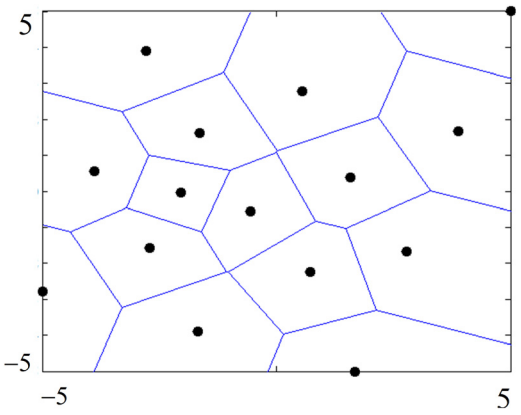

Fig. 1 A set of 2D samples and the corresponding Voronoi cells

$C = \{C_1, C_2, \ldots, C_m\}$ according to the sample set P. The ith Voronoi cell, C_i, represents the surrounding region of point p_i. The cell C_i can be defined as [31]

$$\mathrm{dom}(p_i, p_j) = \left\{ p \in \mathbb{R}^d \middle| \|p - p_i\|_2 \leq \|p - p_j\|_2 \right\} \tag{1}$$

$$C_i = \bigcap_{p_j \in P \setminus p_i} \mathrm{dom}(p_i, p_j) \tag{2}$$

where $\mathrm{dom}(p_i, p_j)$ is a closed half plane bounded by the perpendicular bisector of p_i and p_j. The bisector separates all points of the plane closer to p_i from those closer to p_j. Figure 1 illustrates a 2D example partitioned by the Voronoi diagram approach.

Since a Voronoi cell is a polyhedron with irregular boundaries in high dimensions, it is difficult to describe the cell mathematically. Hence, Monte Carlo method is adopted to generate a large number of random points to describe the Voronoi cells approximately. By computing the distances to all existing sample points, each random point is assigned to its closest point in P. Thus, the Voronoi cell C_i can be approximately described by a set of random points $P^i_{\mathrm{rand}} = \{p^i_{r1}, p^i_{r2}, \cdots, p^i_{rk}\}$, as shown in Fig. 2. This process is described in details in Algorithm 1. The number of random points used in the Monte Carlo simulations is $N(P_{\mathrm{rand}}) = N(P) \times n \times w$, where $N(P)$ is the number of existing points, n is the dimensionality of problems at hand and w is a factor. To investigate the influence of the number of random points on the performance of proposed approach, we use CV-Voronoi approach to generate 100 samples for functions *Peaks* and *Hart6* (see Test Scheme section) with the factor w, respectively, being 100, 200, and 300. It was found that the impact of the factor w on the metamodel accuracy is insignificant in these two cases. Therefore, $w = 100$ is used in this study in order to save computing time.

Algorithm 1. Estimating the Voronoi cells by Monte Carlo approach. P is the current sample set

$P_{\mathrm{rand}} \leftarrow$Random points generated in the design space.
for all $p_r \in P_{\mathrm{rand}}$ **do**
  if the random point p_r is closer to the existing point p_i in P
    Add p_r to the random set P^i_{rand}.
  end
end for

Identify Sensitive Voronoi Cell. In this step, the LOO cross-validation method is utilized to calculate the prediction errors of current samples. The LOO error of point p_i can be calculated as

$$e^i_{\mathrm{LOO}} = \left| f(p_i) - \hat{f}_{P \setminus p_i}(p_i) \right| \tag{3}$$

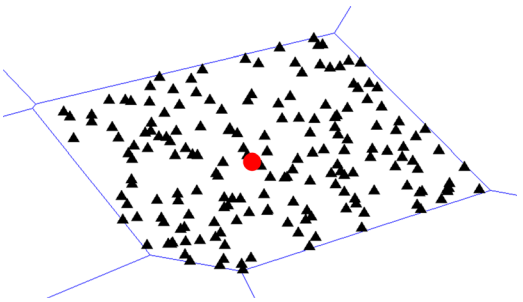

Fig. 2 A Voronoi cell described by a set of random points, where the circle represents the existing sample and the triangles represent the random points

where $f(p_i)$ denotes the real response of p_i, $\hat{f}_{P\setminus p_i}(p_i)$ denotes the predicted response of p_i by the metamodel constructed on existing points without p_i. The above process is repeated for all points in P by leaving them out one at a time and finally the prediction errors of all existing samples will be obtained.

Unlike ACE and SFCVT [25,29], the proposed approach does not try to probe the errors of unobserved points by constructing an error-metamodel. Since the Voronoi cell C_i is a surrounding region of point p_i, it is reasonable to assume that the error behavior of all the points in C_i is similar to that of p_i. Hence, the error behavior of cell C_i can be represented by the prediction error of p_i. Large prediction error of point p_i indicates that the constructed metamodel deviates from the real model largely in cell C_i. Therefore, the prediction error of point p_i can be used to represent the fitting degree in Voronoi cell C_i, which decreases the complexity of the sampling process.

Next, a Voronoi cell with the largest prediction error will be selected as the sensitive Voronoi cell denoted by $C_{\text{sensitive}}$. In this cell, the predicted responses are largely different from the real responses in current stage. Therefore, in order to improve the metamodel accuracy efficiently, more points should be sampled in this sensitive cell. Figure 3 shows the sensitivity analysis of a 2D example, in which the bigger circle implies the larger prediction error.

Choose New Sample. In this step, a new sample p_{new} is sampled in the cell $C_{\text{sensitive}}$. Since a Voronoi cell can be approximately described by a set of random points according to Algorithm 1, the sampling process can be sped up by picking a point from the corresponding random set $P_{\text{rand}}^{\text{sensitive}}$ of $C_{\text{sensitive}}$ based on a certain criterion. In order to obtain the information about the real model as much as possible in $C_{\text{sensitive}}$, the new point and the existing points around it should evenly spread in $C_{\text{sensitive}}$, which can be obtained by the maximin criterion. Since all the points falling in $C_{\text{sensitive}}$ are closer to $p_{\text{sensitive}}$, the process can be simplified as picking a random point that is the farthest away from $p_{\text{sensitive}}$ in $P_{\text{rand}}^{\text{sensitive}}$. This farthest random point is selected as the next sample p_{new} to be evaluated.

Step-by-Step Description for CV-Voronoi. The detailed description about CV-Voronoi is provided as follows:

Given a set of initial points $P = \{p_1, p_2, \ldots, p_m\} \in \mathbb{R}^d$ and the corresponding function responses, $F = \{f(p_1), f(p_2), \ldots, f(p_m)\}$

Begin:

Step 1: Construct a metamodel $\hat{f}$ based on the initial points P and their responses F;

Step 2: According to the current sample set P, Voronoi diagram is used to partition the entire design space into a set of Voronoi cells $C = \{C_1, C_2, \ldots, C_m\}$, each of which is approximately described by a set of random points according to Algorithm 1;

Step 3: Identify the sensitive Voronoi cell $C_{\text{sensitive}}$ by the LOO cross-validation approach

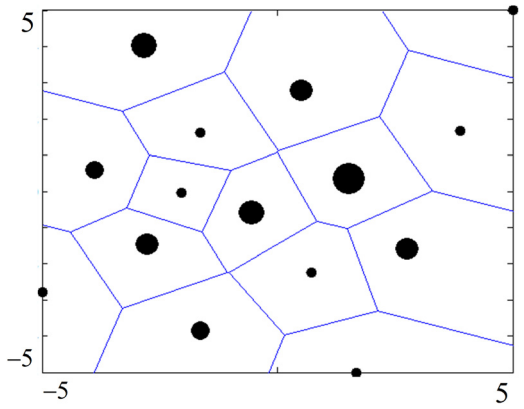

Fig. 3 The prediction errors of some 2D samples and the corresponding Voronoi cells, where bigger circle implies larger error

Step 4: Choose a new sample p_{new} farthest away from $p_{\text{sensitive}}$ from the corresponding random set $P_{\text{rand}}^{\text{sensitive}}$ of $C_{\text{sensitive}}$;

Step 5: Evaluate the response $f(p_{\text{new}})$ of the new point p_{new} and then $P = P \cup \{p_{\text{new}}\}$; $F = F \cup \{f(p_{\text{new}})\}$; $m = m + 1$. Construct and update the metamodel $\hat{f}$;

Step 6: Check the stopping criterion. If yes, stop the algorithm, otherwise, return to Step 2.

End

In Step 1, in order to provide the information about the real function as much as possible, initial samples should spread over the entire design space as evenly as possible. Many space-filling sampling approaches reviewed before can be used here. Since the Kriging technique is particularly suitable for complex functions, it is employed to construct the metamodel in Step 1. The MATLAB toolbox *BlindDACE* [32] is employed to implement the Kriging method for metamodeling in this study.

In Step 6, various stopping criteria may be adopted in CV-Voronoi, such as the root mean square error (RMSE), the maximum absolute error, the CV, the limited number of simulation runs, or the limited computing resource. In this study, we employ RMSE to access the metamodel accuracy, that is defined as

$$\text{RMSE} = \sqrt{\frac{1}{t}\sum_{i=1}^{t}\left(f(q_i) - \hat{f}(q_i)\right)^2} \tag{4}$$

where $f(.)$ is the true function, $\hat{f}(.)$ is a metamodel built on the current samples, t represents the number of test points (here, $t = 5000$) and q_i is a point in the test set. A smaller RMSE implies a more accurate metamodel.

Discussion About CV-Voronoi. As mentioned earlier, ACE and SFCVT [25,29] try to find a new sample with the largest prediction error on the error-metamodel $\hat{f}_{\text{e}} = \hat{f}(p_1, p_2, \ldots, p_m; e_{\text{LOO}}^1, e_{\text{LOO}}^2, \ldots, e_{\text{LOO}}^m)$ under an Euclidean-distance-based constraint. Its optimization process can be described as

$$\begin{aligned} &\text{Find } p_{\text{new}} = \max \hat{f}_{\text{e}}(p) \\ &\text{s.t. } \|p - p_i\|_2 \geq d \ \ \forall p_i \in P \end{aligned} \tag{5}$$

For ACE, the critical distance d is defined as

$$\begin{aligned} d(p_i) &= \min\left(\|p_i - p_j\|_2\right), \ \ \forall p_i \in P \cap (i \neq j) \\ d_{\text{ACE}} &= 0.5 \times \text{mean}(d(p_i)), \ \forall p_i \in P \end{aligned} \tag{6}$$

For SFCVT, it is

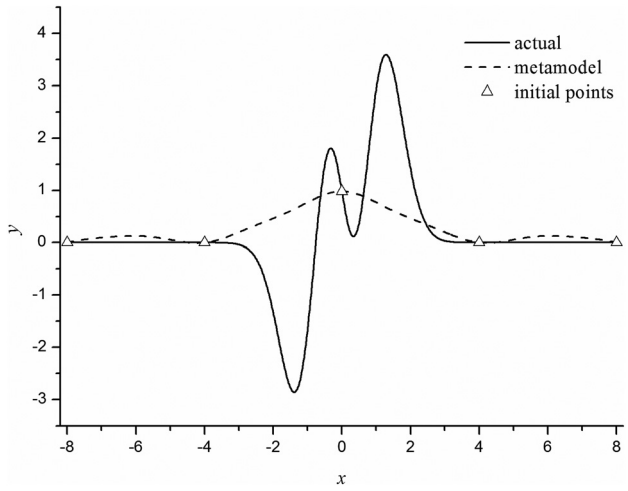

Fig. 4 A 1D function and a metamodel through five initial samples

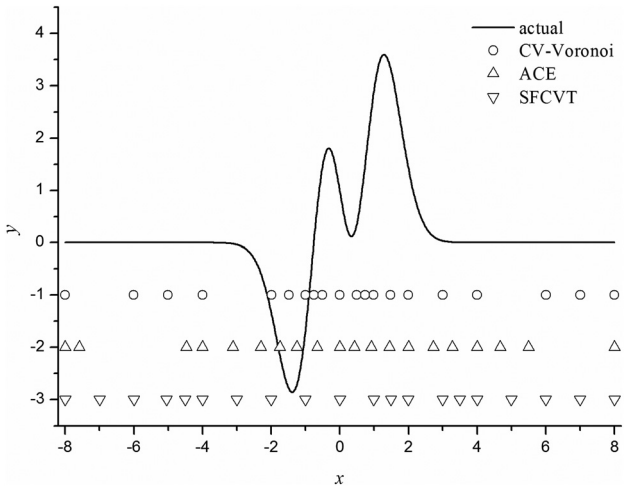

Fig. 5 Samples generated by CV-Voronoi, ACE, and SFCVT for the 1D test case

$$d(p_i) = \min\left(\left\|p_i - p_j\right\|_2\right), \quad \forall p_i \in P \cap (i \neq j)$$
$$d_{\mathrm{SFCVT}} = 0.5 \times \max(d(p_i)), \ \forall p_i \in P \tag{7}$$

The distance constraint prevents new samples from clustering around existing samples. It is observed that d_{SFCVT} is larger than d_{ACE} in each iteration, that leads the samples generated by SFCVT to distribute more sparsely than those obtained by ACE. In practice, various definitions of d may be used for different problems in order to obtain a set of optimal solutions. In general, too large d may force samples to distribute evenly in the design space. As a result, the sampling approach may lose the ability to focus on interesting regions and could not improve the metamodel accuracy rapidly. Too small d may result in that new samples cluster around existing samples, which is, also, against the improvement of metamodel accuracy.

Compared with ACE and SFCVT, the proposed CV-Voronoi approach always samples the region with the largest prediction error, which rapidly reduces the metamodel error. Another advantage of the proposed approach lies in that no parameters need to be defined in advance, that reduces the complexity of the sampling process and makes the approach user-friendly.

A one-dimensional function $y = 3(1-x^2)\mathrm{e}^{(-x^2-1)} - (2x - 10x^3)\mathrm{e}^{-x^2}$ in the range $[-8\ 8]$ is used to illustrate the proposed approach. Figure 4 shows the real function and a Kriging model through five evenly distributed initial points.

It is observed that the real function has several local optima around the origin and the rest regions of which are almost zeros. For this test case, each of the CV-Voronoi, ACE, and SFCVT approaches is used to generate 20 samples with the aforementioned five initial samples, as shown in Fig. 5. It is found that CV-Voronoi adaptively samples more points on the multimodal region while fewer points on the flat region. ACE prefers to pay much attention to the highly nonlinear region, while SFCVT prefers to explore the entire design space.

We introduce a parameter S to reveal the sampling mechanism of the three cross-validation-based sequential sampling approaches. This parameter is defined as the minimal distance from the obtained new point to the existing points during each iteration, which is mathematically described as

$$S = \min\ d(p_{\mathrm{new}}, p_i), \quad p_i \in P \tag{8}$$

The values of S in each iteration step for CV-Voronoi, ACE, and SFCVT with the 1D test case are illustrated in Fig. 6. It is observed that the S value of ACE keeps decreasing during the sampling process. It indicates that ACE keeps exploiting the

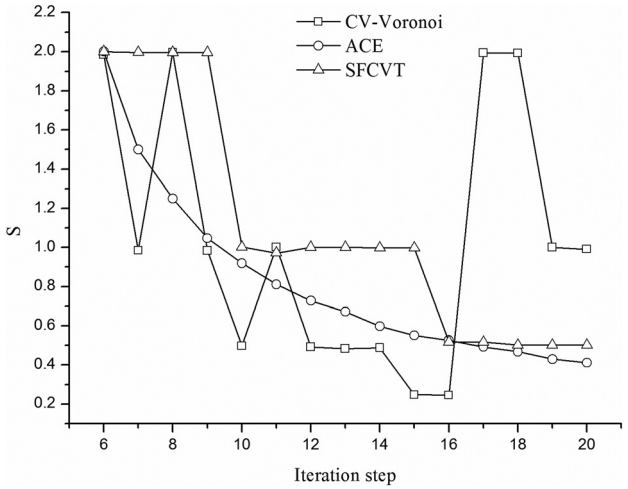

Fig. 6 The *S* values of CV-Voronoi, ACE, and SFCVT during the sampling process

interesting regions so that the sampling density of such regions is increasing and, thus, the new point gets closer to its neighbors. In terms of exploration and exploitation, ACE gives priority to exploitation. It is found that the S value of SFCVT decreases in a ladder-shaped way. The S remains the same during some steps, that means in these steps, SFCVT tries to sample the regions with relatively low sampling density in order to satisfy the space-filling constraint. The steep decrease in S indicates that SFCVT starts sampling the design space at a refined level. In terms of exploration and exploitation, the strict space-filling constraint involved in SFCVT leads this approach to give priority to exploration. Note that the S value of CV-Voronoi fluctuates during the sampling process. When S continues to decrease, CV-Voronoi is exploiting the sensitive region in current stage. When S suddenly increases, it means the previous sensitive region contains enough points so that the error of this region in current stage becomes smaller than another region. Consequently, the proposed approach conveys its attention to another region.

Therefore, due to the error-pursuing mechanism, the proposed approach exploits locally by the identification of sensitive region and explores globally by the shift of sensitive region, as shown in Fig. 7. Note that the sampling process of the proposed approach is executed repeatedly at different levels. For instance, at one-stage, a Voronoi cell, like $\mathrm{Cell_B}$ in Fig. 7, is marked as sensitive region

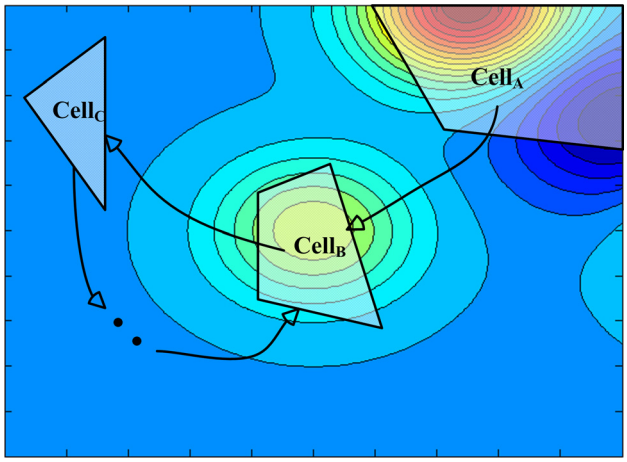

Fig. 7 Illustration of local exploitation and global exploration of the proposed approach

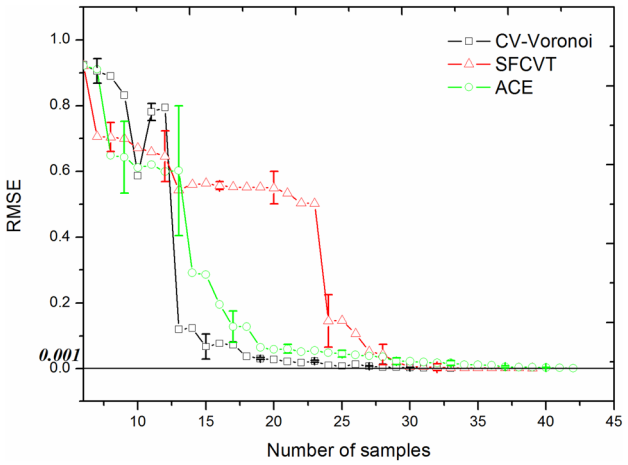

Fig. 9 Convergence histories (mean + SD) of CV-Voronoi, SFCVT, and ACE for a 1D test case

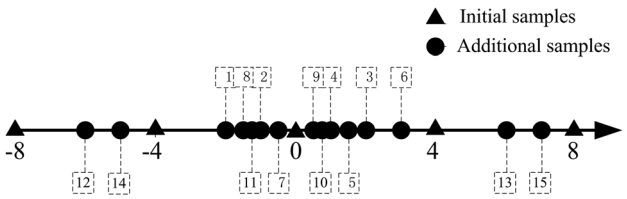

Fig. 8 The trail of the additional 15 samples for the aforementioned 1D case by the proposed approach

and then new samples are sampled in this cell. After adding new samples, if the prediction error of this cell is smaller than that of another cell, the proposed approach will move to another cell even if Cell$_B$ may have not been fitted well yet. This new sensitive region may even be a relatively flat region like Cell$_C$ in Fig. 7. After several steps, the Voronoi cell Cell$_B$ may be identified as a sensitive region again to be resampled. Such a gradually refined process enables the proposed approach to improve the metamodel accuracy globally. To give an intuitive view of the local exploitation and global exploration of the proposed approach, Fig. 8 shows the trail of the additional 15 samples generated by the proposed approach in the aforementioned 1D case. It is found that the proposed approach does not fall into local exploitation. After 11 steps, the approach jumps out of the multimodal regions and places several samples in those flat boundary regions.

To further confirm the discussions above, we apply the three approaches for the 1D test case to reach a specified root mean square error (i.e., RMSE = 0.001) with the aforementioned five initial samples. Each experiment runs 10 times in the MATLAB environment in order to avoid unrepresentative numerical results. The metamodel performance of each approach is illustrated in Fig. 9. It is observed that CV-Voronoi outperforms the other two approaches. It requires only 33 samples to reach the desired metamodel accuracy, while SFCVT needs 39 samples and ACE needs 42 samples to reach the same RMSE. Moreover, from Fig. 9, it can be observed that for a large critical RMSE (e.g., RMSE $\geq$ 0.6), CV-Voronoi does not outperform SFCVT and ACE. But at this RMSE level, the constructed metamodel differs largely from the real model, which is useless in practice. In later stage with small critical RMSE, CV-Voronoi improves the metamodel accuracy quickly. It is also observed that because of the emphasis on exploitation, ACE converges faster than SFCVT when RMSE $\geq$ 0.02. However, when targeting toward a smaller critical RMSE value, SFCVT converges much faster than ACE.

Examples and Discussions

Test Scheme. In this section, eight test cases with different features are used to demonstrate the applicability of the CV-Voronoi approach. These test cases are classified into two groups: the first group is a set of test cases with both flat and highly nonlinear regions. These test cases are in favor of sequential sampling methods considering both exploration and exploitation; the second group is a set of test cases with uniform behavior. These test cases are in favor of exploration-based (space-filling) sequential sampling methods. The eight test cases are summarized in Table 1 and their mathematical functions are listed below.

Peaks function

$$y = 3(1-x_1)^2 e^{\left(-x_1^2-(x_2+1)^2\right)} - 10\left(\frac{x_1}{5} - x_1^3 - x_2^5\right)e^{\left(-x_1^2-x_2^2\right)} - \frac{1}{3}e^{\left(-(x_1+1)^2-x_2^2\right)}, \quad x_{1,2} \in [-5,5] \tag{9}$$

Easom function

$$y = -\cos(x_1)\cos(x_2)\exp(-(x_1-\pi)^2-(x_2-\pi)^2), \quad x_{1,2} \in [0,7] \tag{10}$$

Hartman function with 3 variables (Hart3)

$$y = -\sum_{i=1}^{4} c_i \exp\left[-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right], \quad x_j \in [0,1], j = 1,2,3$$

where

$$[a_{ij}]_{j=1,\ldots,6} = \begin{bmatrix} 3.0 & 10 & 30 \\ 0.1 & 10 & 35 \\ 3.0 & 10 & 30 \\ 0.1 & 10 & 36 \end{bmatrix} \tag{11}$$

$$c_i = [1 \quad 1.2 \quad 3 \quad 3.2]^T$$

$$[p_{ij}]_{j=1,\ldots,6} = \begin{bmatrix} 0.3689 & 0.1170 & 0.2673 \\ 0.4699 & 0.4387 & 0.7470 \\ 0.1091 & 0.8732 & 0.5547 \\ 0.0381 & 0.5743 & 0.8828 \end{bmatrix}$$

Shekel function

Table 1 Summary of test cases

Group	Test function	Range	Number of variables	Performance
Group 1	Peaks	$[-5\ 5]^2$	2	High nonlinearity in the center region
	Easom	$[0\ 7]^2$	2	Nonlinearity in the center region
	Hart3	$[0\ 1]^3$	3	High dimension with high nonlinearity
	Shekel	$[3\ 5]^4$	4	High dimension with multimodal
	Hart6	$[0\ 1]^6$	6	High dimension with high nonlinearity
	Ackley10	$[-0.6\ 0.6]^{10}$	10	High dimension with high nonlinearity
Group 2	Shubert	$[1\ 3]^2$	2	Waving uniformly
	Rastrigin	$[0\ 1]^2$	2	Waving uniformly

$$y = \sum_{i=1}^{5} \frac{1}{c_i + \sum_{j=1}^{4} (x_j - a_{ij})^2}, \quad x_i \in [3\ 5], \quad i = 1, 2, 3, 4$$

where

$$a = \begin{bmatrix} 4.0 & 4.0 & 4.0 & 4.0 \\ 1.0 & 1.0 & 1.0 & 1.0 \\ 8.0 & 8.0 & 8.0 & 8.0 \\ 6.0 & 6.0 & 6.0 & 6.0 \\ 3.0 & 7.0 & 3.0 & 7.0 \end{bmatrix} \text{ and } c = \begin{bmatrix} 0.1 \\ 0.2 \\ 0.2 \\ 0.4 \\ 0.6 \end{bmatrix} \tag{12}$$

Hartman function with 6 variables (Hart6)

$$y = -\sum_{i=1}^{4} c_i \exp\left[-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right], x_j \in [0, 1], j = 1, \ldots, 6$$

where

$$[a_{ij}]_{j=1,\ldots,6} = \begin{bmatrix} 10 & 3 & 17 & 3.5 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{bmatrix}$$

$$c_i = [1 \quad 1.2 \quad 3 \quad 3.2]^T \tag{13}$$

$$[p_{ij}]_{j=1,\ldots,6} = \begin{bmatrix} 0.1312 & 0.1696 & 0.5569 & 0.0124 & 0.8283 & 0.5886 \\ 0.2329 & 0.4139 & 0.8307 & 0.3736 & 0.1004 & 0.9991 \\ 0.2348 & 0.1451 & 0.3522 & 0.2883 & 0.3047 & 0.6550 \\ 0.4047 & 0.8828 & 0.8732 & 0.5743 & 0.1091 & 0.0381 \end{bmatrix}$$

Ackley function with 10 variables (Ackley10)

$$y = -20\exp\left(-0.2\sqrt{\frac{1}{10}\sum_{i=1}^{10} x_i^2}\right) - \exp\left(\frac{1}{10}\sum_{i=1}^{10}\cos(2\pi x_i)\right) + 20 + \exp(1)$$
$$x_i \in [-0.6, 0.6], i = 1, 2, \ldots, 10 \tag{14}$$

Shubert function

$$y = \left(\sum_{i=1}^{5} i\cos((i+1)x_1 + i)\right) \times \left(\sum_{i=1}^{5} i\cos((i+1)x_2 + i)\right), x_{1,2} \in [1, 3] \tag{15}$$

Rastrigin function

$$y = 20 + \sum_{i=1}^{2} \left(x_i^2 - 10\cos(2\pi x_i)\right), x_{1,2} \in [0, 1] \tag{16}$$

In this study, CV-Voronoi is compared with LOLA-Voronoi [30], SFCVT [25], MIPT [22], and MSE [12] which have shown their merits in other studies. Note that Aute et al. [29] has demonstrated that SFCVT outperforms ACE. Therefore, ACE is not investigated in this comparison.

In this article, we use the TPLHD approach [10] with a one-point seed to generate the initial design for all test cases. Samples generated by TPLHD cover the entire design space with both good space-filling and projective properties. The number of initial samples is set to $5 \times n$ (n represents the number of variables) and sensitive analysis about the influence of the number of initial points on the performance of the proposed approach will be studied later.

The test scheme is to obtain the sample size required to reach the desired metamodel accuracy by using different sequential sampling approaches. The sampling process stops when the RMSE of a metamodel is below a prespecified value or the iteration reaches to the maximum number of function evaluations (i.e., 1000 in this study). In this test, each case will run 10 times in the MATLAB environment in order to avoid unrepresentative numerical results.

Test Results and Discussion. Table 2 shows the results of all test cases using different sequential sampling approaches. The third column in Table 2 lists the critical RMSE for each case. For the low dimensional cases (less than or equal to 5 variables), the critical RMSE is required to less than or equal to 0.06. Metamodels at this RMSE level are thought to be a good approximation of the real functions in practice. For functions with pretty small responses (e.g., *Easom*), we set a lower critical RMSE. For the high dimensional cases (more than 5 variables), the critical RMSE is set to be a higher value than that for the low dimensional cases, i.e., 0.10 for function *Hart6* and 0.30 for function *Ackley10*. This takes into account the challenge of improving the metamodel accuracy in high dimensional space. The results tabulated in Table 2 are the average values from the 10 runs of sampling required to reach the critical RMSE, and the standard deviation for each sampling approach.

For group 1 with both flat and highly nonlinear regions, they are in favor of sequential sampling methods considering both exploration and exploitation. It is observed that CV-Voronoi outperforms other approaches in all the test cases of group 1 by achieving the desired metamodel accuracy with the smallest sample size, especially for functions *Peaks* and *Hart6*. According to the discussions before, because of the strict constraint function, the performance of SFCVT is similar to that of a space-filling sequential sampling approach. It is found that SFCVT provides the worst performance in the first three adaptive sampling approaches.

It is observed that LOLA-Voronoi performs worse than MIPT and MSE in function *Easom*. This is mainly due to the reason that

Table 2 Summary of test results

Group	Test function	RMSE	CV-Voronoi	LOLA-Voronoi	SFCVT	MIPT	MSE
Group 1	Peaks	0.03	**133.7 ± 10.9**	181.6 ± 13.0	203.1 ± 16.6	262.4 ± 18.2	319.7 ± 20.5
	Easom	0.005	**79.0 ± 5.1**	129.9 ± 6.3	81.4 ± 10.5	87.5 ± 10.0	113.2 ± 5.7
	Hart3	0.05	**90.2 ± 8.4**	116.6 ± 11.0	143.0 ± 18.4	121.8 ± 10.4	98.2 ± 3.1
	Shekel	0.06	**631.7 ± 33.6**	>1000	>1000	725.3 ± 34.1	>1000
	Hart6	0.10	**449.6 ± 74.1**	516.6 ± 102.7	710.6 ± 68.8	902.8 ± 74.8	>1000
	Ackley10	0.30	**923.6 ± 68.9**	>1000	>1000	>1000	>1000
Group 2	Shubert	0.05	104.2 ± 2.5	118.3 ± 6.0	114.9 ± 12.1	113.5 ± 4.5	**95.4 ± 1.3**
	Rastrigin	0.05	98.4 ± 5.4	129.4 ± 3.3	123.2 ± 6.7	**95 ± 4.1**	98.0±7.8

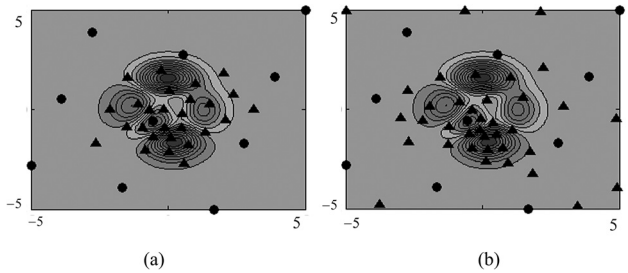

Fig. 10 Two runs for function *Peaks* with (*a*) CV-Voronoi (37 samples) and (*b*) LOLA-Voronoi (47 samples), separately

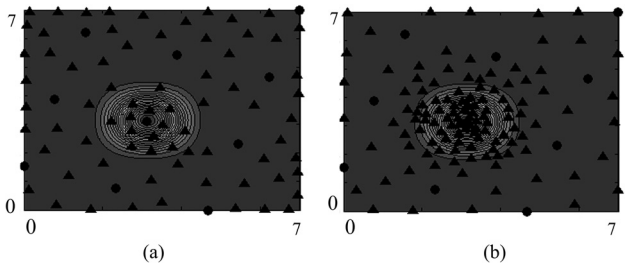

Fig. 11 Two runs for function *Easom* with (*a*) CV-Voronoi (83 samples) and (*b*) LOLA-Voronoi (126 samples), separately

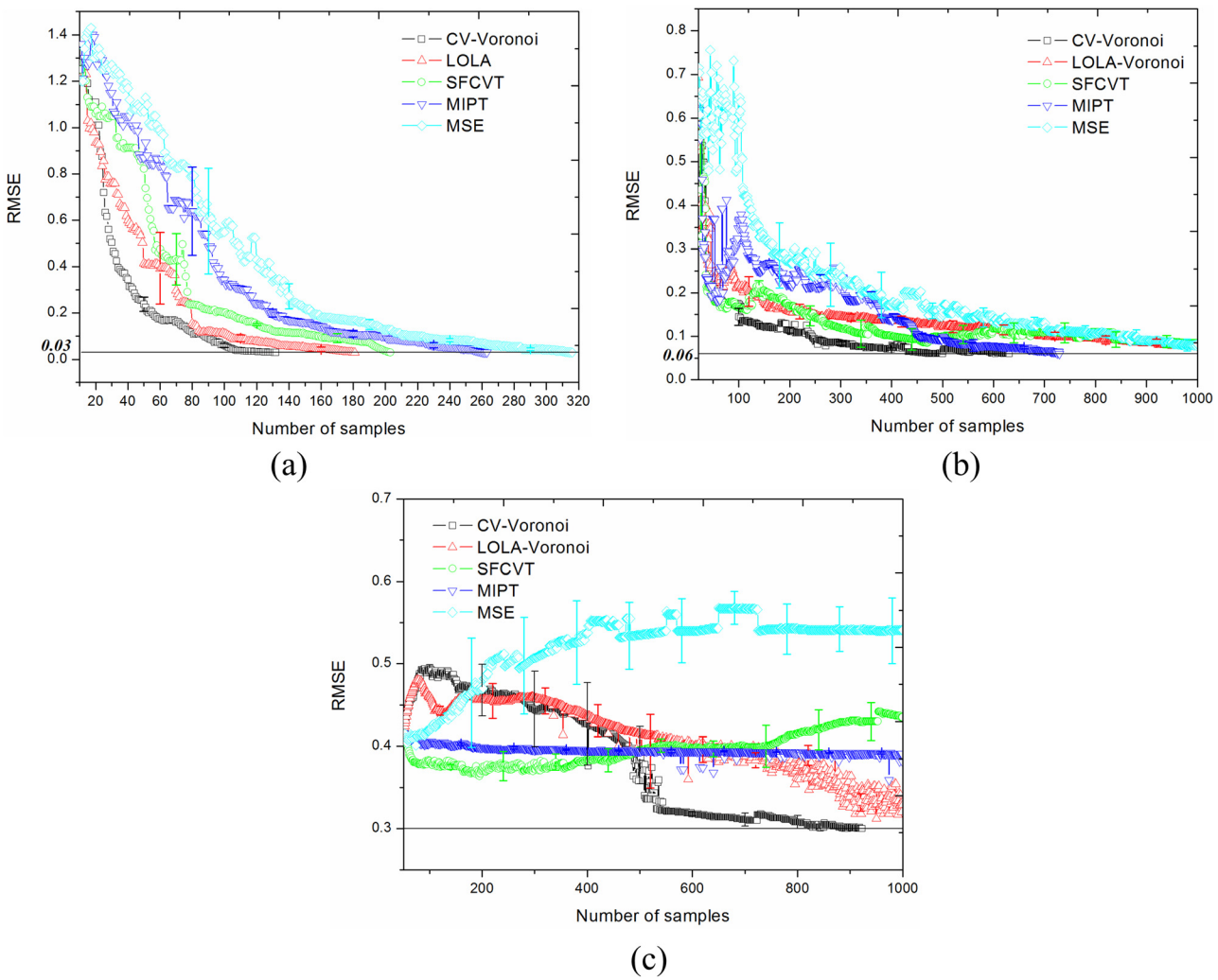

Fig. 12 Convergence histories (mean + SD) of different approaches for functions: (*a*) *Peaks*, (*b*) *Shekel*, and (*c*) *Ackley10*.

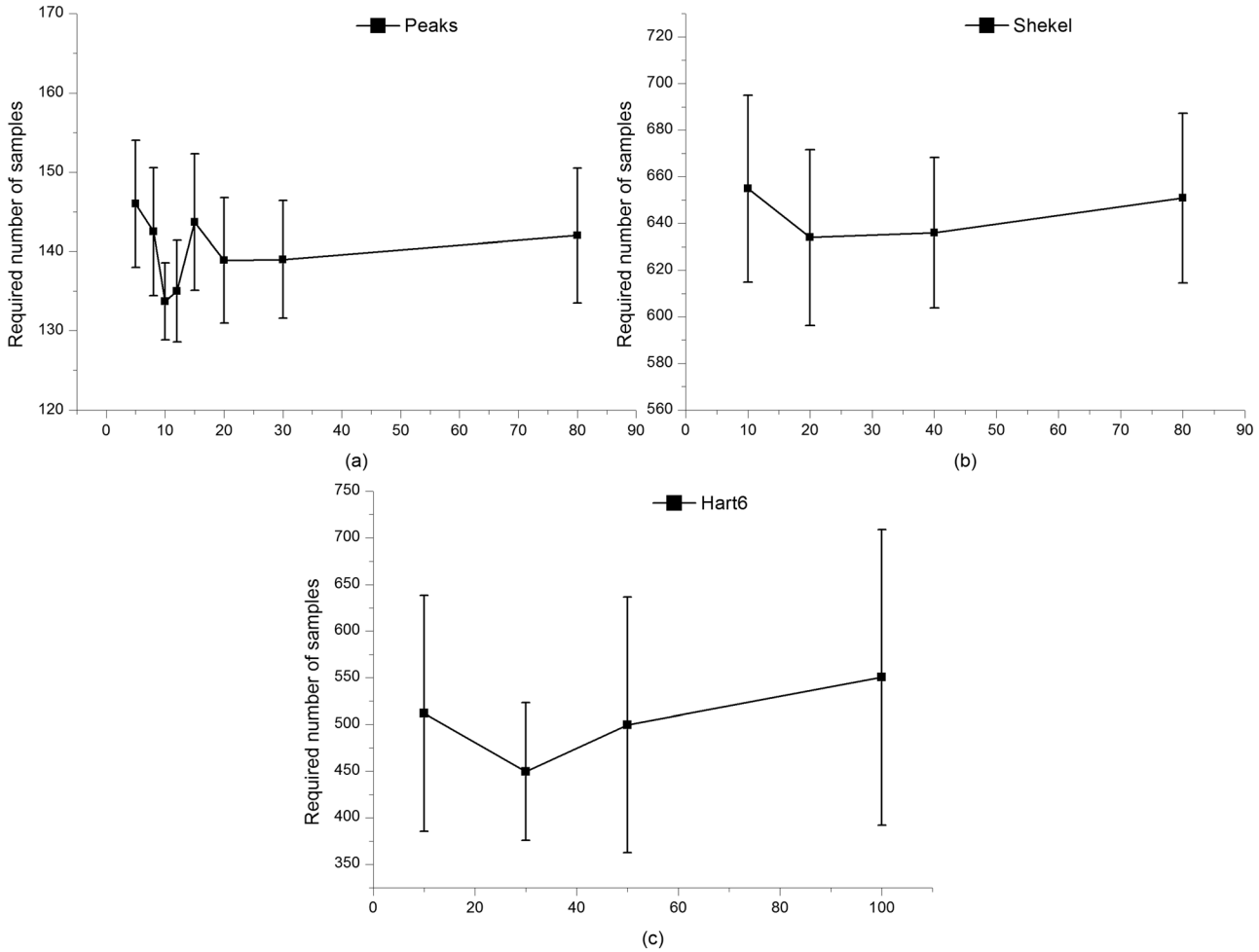

Fig. 13 Required number of samples (mean + SD) to reach specified RMSE by CV-Voronoi with different numbers of initial samples for functions (*a*) *Peaks*, (*b*) *Shekel*, and (*c*) *Hart6*.

LOLA-Voronoi gives fixed weights to exploration and exploitation, which could not change with problems at hand. To highlight the rigidity of LOLA-Voronoi, the function *Peaks* with two different prespecified critical values of RMSE is further investigated in this study. In the first situation, the prespecified RMSE value for function *Peaks* is roughly set to 0.5. Considering that the region close to the origin of *Peaks* contains several local optima while the rest regions are almost zeros, a rapid way to improve the metamodel accuracy in this situation is to concentrate on exploiting the highly nonlinear region and ignore the flat region temporarily. Figure 10, respectively, illustrates a run from CV-Voronoi and LOLA-Voronoi, where circles represent the initial samples and triangles represent the new samples. It is observed that CV-Voronoi achieves the desired RMSE with only 33 samples by giving priority to exploitation, whereas LOLA-Voronoi requires nearly 51% more samples to reach the desired value of RMSE because of the rigid treatment of exploration and exploitation. In the second situation, the prespecified RMSE value for function *Peaks* is set to 0.03. In this situation, the early stage of sampling process should give priority to exploitation, which is similar to the first situation, while in the later stage, since the highly nonlinear region has been fitted relatively well, the priority should be given to exploration. The proposed error-pursuing approach samples more points on the multimodal region while still covering the flat region with relatively sparse points. As a result, CV-Voronoi achieves the desired RMSE with 134 samples, whereas LOLA-Voronoi requires nearly 36% more samples.

The results of function *Easom* offer the further demonstration about the shortcomings of LOLA-Voronoi. Figure 11, respectively, picks a run from CV-Voronoi and LOLA-Voronoi for function *Easom*. Note that *Easom* is a single-modal function and the nonlinearity of which is smaller than that of the multimodal function *Peaks*. As we can see, LOLA-Voronoi puts emphasis on exploiting the single-modal region, that results in the largest sample size to reach the desired RMSE in this case. Compared with LOLA-Voronoi, CV-Voronoi does not have an explicit expression for describing the relationship between exploration and exploitation. However, the comparative results show that the proposed approach provides the best performance in this study.

It is observed that for test cases in group 1, MIPT provides better performance than MSE. It is mainly because MIPT generates samples with both good space-filling and projective properties, while MSE only considers the space-filling property under the Bayesian framework. Since samples generated by MSE have the property of clustering on the boundary of design space in high dimensions, MSE provides the worst performance in functions *Shekel*, *Hart6*, and *Ackley10*.

To further verify the discussions above, we plot the convergence histories of different approaches for functions *Peaks*, *Shekel*, and *Ackley10* in Fig. 12. It is observed that CV-Voronoi converges faster than other approaches. For function *Shekel*, LOLA-Voronoi, SFCVT, and MSE improve the metamodel accuracy so slowly in the later stage that they could not reach the required RMSE within the limited number of function evaluations. For function *Ackley10*, it is difficult to improve the metamodel accuracy in such a high dimensional domain. No approach except CV-Voronoi can reach the required RMSE within the limited computing resource. Note that because the MSE approach

samples more points on the boundary of the design space, it could not effectively improve the metamodel accuracy for this high dimensional case.

Moreover, from the convergence histories in Fig. 12, we can observe the influence of the critical RMSE value on the performance of each approach. In the early stage with a large critical RMSE value (e.g., for *Shekel,* large critical RMSE means RMSE ≥ 0.2), each approach converges with much fluctuation. Obviously, at this RMSE level, the constructed metamodel is coarse and, thus, is meaningless in practice. In later stage with a small critical RMSE value, the test results show that the proposed approach provides the most efficient way to improve the metamodel accuracy from low to high dimensions.

For test cases in group 2, since the behaviors of these test cases are similar everywhere, emphasis should be placed on exploration in order to improve the metamodel accuracy efficiently. As expected, MIPT and MSE have the best performance in test cases of group 2 since they sample the design space evenly. Note that the results of CV-Voronoi are very close to those of MIPT and MSE, which shows a good adaptability for problems at hand.

Based on the comparative results, it can be concluded that CV-Voronoi outperforms LOLA-Voronoi and SFCVT. CV-Voronoi performs significantly better than the space-filling sampling approaches, such as MIPT and MSE in test cases of group 1, and has the comparable performance to MIPT and MSE in test cases of group 2.

Sensitive Analysis of the Number of Initial Samples. An important factor influencing the efficiency of sampling approaches is the number of initial samples. Proper number of initial samples actually depends on problems at hand. Some problems may require few initial samples (thus allowing for more adaptively sampled points) while others may require more. Generally, it is suggested that the number of initial samples should depend on the dimension of problems.

To study the impact of the number of initial samples on the performance of the proposed approach, we applied CV-Voronoi approach with different numbers of initial samples for functions *Peaks*, *Shekel*, and *Hart6* to reach the specified RMSE listed in Table 2. Each experiment runs for 10 times. Test results shown in Fig. 13 indicate that the number of initial samples ($5 \times n$) adopted in this study is proper for these three cases. In addition, it is observed that for these three cases, too few initial samples bring few information and, thus, lead the proposed approach to converge slowly, while too many initial samples make the majority of the samples not chosen adaptively, which, also, leads the proposed approach to require more samples to reach the desired metamodel accuracy.

Conclusions

In this paper, we propose a novel, simple, and robust cross-validation-based sequential sampling approach that adaptively improves the metamodel accuracy by the error-pursuing mechanism. First, it employs the Voronoi diagram approach to partition the entire design space into a set of Voronoi cells according to the current samples, which helps us to concentrate on several regions of interest rather than the entire design space. Second, the proposed approach calculates the prediction errors of current samples by the leave-one-out cross-validation approach. Since any point falls into a Voronoi cell C_i is closer to the existing point p_i, the prediction error of p_i can be used to simply represent the error behavior of C_i. Then the proposed approach identifies the sensitive region with the largest prediction error in current iteration. Finally, a new point is sampled in the obtained sensitive region based on the maximin criterion. During the sampling process, the CV-Voronoi approach exploits locally by the identification of sensitive region and explores globally with the shift of sensitive region.

A comparison study has been conducted to investigate the proposed method with LOLA-Voronoi, SFCVT, MIPT, and MSE on eight test cases with different features. The test results have demonstrated that the proposed approach is robust and efficient. It requires the smallest sample size to reach the desired metamodel accuracy for test cases of group 1, while having a comparable performance to MIPT and MSE for test cases of group 2.

The only information utilized in CV-Voronoi to guide the sampling process is the existing samples and the previously constructed metamodels. Hence, CV-Voronoi is simple and runs in a compact way. It can run with any metamodel technique to improve the accuracy of global metamodeling efficiently, which is meaningful for simulation-based engineering problems.

Acknowledgment

The authors appreciate the financial support from Ph.D. Programs Foundation of Liaoning Province (20131019), National Natural Science Foundation of China (51308090), The Fundamental Research Funds for the Central Universities (DUT10RC(3)109) and National Basic Research Program of China (2009CB724303).

Nomenclature

C = a set of Voronoi cells
C_i = the corresponding Voronoi cell of point p_i
$C_{\text{sensitive}}$ = the identified sensitive Voronoi cell
e^i_{LOO} = the prediction error of p_i
f = the real model
$\hat{f}$ = the metamodel of f
$N(P)$ = number of samples in P
P = the sample set
p_i = an existing sample in P
p_{new} = a new sample obtained by the proposed approach in one iteration
$p_{\text{sensitive}}$ = the existing point contained in $C_{\text{sensitive}}$
P^i_{rand} = a set of random samples used to describe C_i
$P^{\text{sensitive}}_{\text{rand}}$ = the corresponding random set of the sensitive Voronoi cell $C_{\text{sensitive}}$

References

[1] Cressie, N., 1988, "Spatial Prediction and Ordinary Kriging," Math. Geol., **20**(4), pp. 405–421.
[2] Dyn, N., Levin, D., and Rippa, S., 1986, "Numerical Procedures for Surface Fitting of Scattered Data by Radial Functions," SIAM J. Sci. Stat. Comput., **7**(2), pp. 639–659.
[3] Fang, H., and Horstemeyer, M. F., 2006, "Global Response Approximation With Radial Basis Functions," Eng. Optim., **38**(4), pp. 407–424.
[4] Friedman, J. H., 1991, "Multivariate Adaptive Regression Splines," Ann. Stat., **19**(1), pp. 1–67.
[5] Wang, G. G., and Shan, S., 2007, "Review of Metamodeling Techniques in Support of Engineering Design Optimization," ASME J. Mech. Des., **129**(4), pp. 370–380.
[6] McKay, M. D., Beckman, R. J., and Conover, W. J., 1979, "Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code," Technometrics, **21**(2), pp. 239–245.
[7] Owen, A. B., 1992, "Orthogonal Arrays for Computer Experiments, Integration and Visualization," Stat. Sin., **2**(2), pp. 439–452.
[8] Ye, K. Q., Li, W., and Sudjianto, A., 2000, "Algorithmic Construction of Optimal Symmetric Latin Hypercube Designs," J. Stat. Plann. Inference, **90**(1), pp. 145–159.
[9] Grosso, A., Jamali, A., and Locatelli, M., 2009, "Finding Maximin Latin Hypercube Designs by Iterated Local Search heuristics," Eur. J. Oper. Res., **197**(2), pp. 541–547.
[10] Viana, F. A., Venter, G., and Balabanov, V., 2010, "An Algorithm for Fast Optimal Latin Hypercube Design of Experiments," Int. J. Numer. Methods Eng., **82**(2), pp. 135–156.
[11] Zhu, H., Liu, L., Long, T., and Peng, L., 2012, "A Novel Algorithm of Maximin Latin Hypercube Design Using Successive Local Enumeration," Eng. Optim., **44**(5), pp. 551–564.
[12] Jin, R., Chen, W., and Sudjianto, A., 2002, "On Sequential Sampling for Global Metamodeling in Engineering Design," Proceedings of ASME Design Automation Conference, Montreal, Canada, September 29–October 2, 2002, ASME, pp. 539–548.
[13] Jones, D. R., Schonlau, M., and Welch, W. J., 1998, "Efficient Global Optimization of Expensive Black-Box Functions," J. Global Optim., **13**(4), pp. 455–492.

[14] Booker, A. J., Dennis, J., Jr., Frank, P. D., Serafini, D. B., Torczon, V., and Trosset, M. W., 1999, "A Rigorous Framework for Optimization of Expensive Functions by Surrogates," Struct. Optim., **17**(1), pp. 1–13.
[15] Cox, D. D., and John, S., 1997, "SDO: A Statistical Method for Global Optimization," In N. Alexandrov and M. Y. Hussaini, *Multidisciplinary Design Optimization: State of the Art*, SIAM, Philadelphia, pp. 315–329.
[16] Trosset, M. W., and Torczon, V., 1997, "Numerical Optimization Using Computer Experiments," NASA Langley Research Center, Technical Report No. 97-38.
[17] Johnson, M. E., Moore, L. M., and Ylvisaker, D., 1990, "Minimax and Maximin Distance Designs," J. Stat. Plann. Inference, **26**(2), pp. 131–148.
[18] Audze, P., and Eglais, V., 1977, "New Approach for Planning Out of Experiments," Prob. Dyn. Strengths, **35**, pp. 104–107.
[19] Fang, K. T., Ma, C. X., and Winker, P., 2002, "Centered L2-Discrepancy of Random Sampling and Latin Hypercube Design, and Construction of Uniform Designs," Math. Comput., **71**(237), pp. 275–296.
[20] Iman, R. L., and Conover, W., 1980, "Small Sample Sensitivity Analysis Techniques for Computer Models. With an Application to Risk Assessment," Commun. Stat. Theory Methods, **9**(17), pp. 1749–1842.
[21] Xiong, F., Xiong, Y., Chen, W., and Yang, S., 2009, "Optimizing Latin Hypercube Design for Sequential Sampling of Computer Experiments," Eng. Optim., **41**(8), pp. 793–810.
[22] Crombecq, K., Laermans, E., and Dhaene, T., 2011, "Efficient Space-Filling and Non-Collapsing Sequential Design Strategies for Simulation-Based Modeling," Eur. J. Oper. Res., **214**(3), pp. 683–696.
[23] Watson, A. G., and Barnes, R. J., 1995, "Infill Sampling Criteria to Locate Extremes," Math. Geol., **27**(5), pp. 589–608.
[24] Farhang Mehr, A., and Azarm, S., 2005, "Bayesian Meta-Modelling of Engineering Design Simulations: A Sequential Approach With Adaptation to Irregularities in the Response Behaviour," Int. J. Numer. Methods Eng., **62**(15), pp. 2104–2126.
[25] Li, G., Aute, V., and Azarm, S., 2010, "An Accumulative Error Based Adaptive Design of Experiments for Offline Metamodeling," Struct. Multidiscip. Optim., **40**(1–6), pp. 137–155.
[26] Forrester, A., Sóbester, A., and Keane, A., 2008, *Engineering Design via Surrogate Modelling: A Practical Guide*, John Wiley & Sons, Chichester, UK.
[27] Lin, Y., 2004, "An Efficient Robust Concept Exploration Method and Sequential Exploratory Experimental Design," Ph.D. thesis, Georgia Institute of Technology.
[28] Turner, C. J., Crawford, R. H., and Campbell, M. I., 2007, "Multidimensional Sequential Sampling for NURBs-Based Metamodel Development," Eng. Comput., **23**(3), pp. 155–174.
[29] Aute, V., Saleh, K., Abdelaziz, O., Azarm, S., and Radermacher, R., 2013, "Cross-Validation Based Single Response Adaptive Design of Experiments for Kriging Metamodeling of Deterministic Computer Simulations," Struct. Multidiscip. Optim., **48**(3), pp. 581–605.
[30] Crombecq, K., Gorissen, D., Deschrijver, D., and Dhaene, T., 2011, "A Novel Hybrid Sequential Design Strategy for Global Surrogate Modeling of Computer Experiments," SIAM J. Sci. Comput., **33**(4), pp. 1948–1974.
[31] Aurenhammer, F., 1991, "Voronoi diagrams—a survey of a fundamental geometric data structure," J. ACM Comput. Surveys, **23**(3), pp. 345–405.
[32] Couckuyt, I., Forrester, A., Gorissen, D., De Turck, F., and Dhaene, T., 2012, "Blind Kriging: Implementation and Performance Analysis," Adv. Eng. Software, **49**, pp. 1–13.