

# An Efficient First-Principles Saddle Point Searching Method Based on Distributed Kriging Metamodels

**Anh Tran**

George W. Woodruff School of  
Mechanical Engineering,  
Georgia Institute of Technology,  
Atlanta, GA 30332  
e-mail: anh.vt2@gatech.edu

**Lijuan He**

George W. Woodruff School of  
Mechanical Engineering,  
Georgia Institute of Technology,  
Atlanta, GA 30332  
e-mail: helijuan130@gmail.com

**Yan Wang<sup>1</sup>**

George W. Woodruff School of  
Mechanical Engineering,  
Georgia Institute of Technology,  
Atlanta, GA 30332  
e-mail: yan.wang@me.gatech.edu

*Searching for local minima, saddle points, and minimum energy paths (MEPs) on the potential energy surface (PES) is challenging in computational materials science because of the complexity of PES in high-dimensional space and the numerical approximation errors in calculating the potential energy. In this work, a local minimum and saddle point searching method is developed based on kriging metamodels of PES. The searching algorithm is performed on both kriging metamodels as the approximated PES and the calculated one from density functional theory (DFT). As the searching advances, the kriging metamodels are further refined to include new data points. To overcome the dimensionality problem in classical kriging, a distributed kriging approach is proposed, where clusters of data are formed and one metamodel is constructed within each cluster. When the approximated PES is used during the searching, each predicted potential energy value is an aggregation of the ones from those metamodels. The dimension of each metamodel is further reduced based on the observed symmetry in materials systems. The uncertainty associated with the ground-state potential energy is quantified using the statistical mean-squared error in kriging to improve the robustness of the searching method.*

[DOI: 10.1115/1.4037459]

## 1 Introduction

Understanding the process–structure–property relation is crucial in designing the materials for engineering problems. Computational tools, such as density functional theory (DFT), molecular dynamics simulation, and finite element methods, can be utilized to predict the material properties based on material structures. Such linkages span a wide range of length and time scales in materials science. Understanding the material behaviors at micro-scale and nanoscale are essential to predict the material properties. For complex material systems which contain multiple chemical components, the material behaviors are even harder to comprehend, because there are many metastable states at which thermodynamically distinct phases occur and coexist. Quantitative description of phases, transition, and transformation processes between phases is the key to improve and create desirable material properties.

At nanoscale, the phase transition process can be predicted from potential energy surface (PES). The PES is a hypersurface in a high-dimensional configurational space that corresponds to the potential energy based on the geometric configurations of the material system. PES can be constructed via calculating potential energy with first-principles approaches such as DFT. In molecular dynamics simulation, the geometric configurations are given by the atomistic positions, velocities, and interatomic forces. The local minima on the PES indicate stable or metastable states of the material system. A minimum energy path (MEP) [1,2] is the lowest energy path that connects multiple local minima and describes how the material system transforms from one metastable state to another. The maximum potential energy along the MEP, which is a saddle point, determines the activation energy which the material system needs to overcome to complete the phase transition process from one phase to another. As a result, the activation energy is obtained by first searching for the MEP on PES, and

then finding saddle points along the MEP. With the activation energy, the transition rate is calculated based on the transition state theory [3].

Dynamic behaviors of materials, including chemical reaction, diffusion, or protein folding, are regarded as phase transitions or transformations at atomistic level. The prediction of phase transition relies on the accurate estimation of activation energy and saddle points on PES. The overestimation of activation energy could lead to underestimation of transition rates in crack propagation, irradiation creep, or chemical erosion in structural materials, which may cause catastrophic structure failures. The inaccurate estimation of transition rates during the design stage could also lead to high transient errors in phase-change memory materials for information storage, or ineffective binding of new drugs, which could result in significant costs and consequences for individuals and society. The risks associated with wrong activation energy estimations are high, as simulation has become a universal engineering tool for materials design.

Searching on the PES is a difficult problem, because the dimension of the configurational space is typically very high. This problem is also referred to as the curse of dimensionality in literature. For complex material systems which contain multiple chemical components, the curse of dimensionality significantly impacts the efficiency and accuracy of the searching process. Specifically, for  $n$  atoms, the dimension of the configurational space is  $3n$ . There are many literature reviews [4–7] on a number of the state-of-the-art saddle points and transition pathway search methods. Some of the most used search methods to find saddle point include nudge elastic bands [8,9], ridge [10], Dewar–Healy–Stewart [11], and dimer [12]. The existing search methods generally can be divided into two groups: single-ended and double-ended methods. The single-ended methods aim to locate the saddle points from initial configuration and do not construct the MEP. The double-ended methods aim to locate the saddle points and the transition pathways, i.e., the MEP connecting the initial and final configurations. The majority of single-ended methods depend on the Hessian eigenvalues and eigenvectors with local the quadratic approximation of PES within a trust region [13]. These methods rely directly

<sup>1</sup>Corresponding author.

Manuscript received January 14, 2017; final manuscript received June 25, 2017; published online September 7, 2017. Assoc. Editor: Sankaran Mahadevan.

on a large number of the functional evaluations, which is more accurate but also computationally expensive.

In DFT calculations, there are many approximations and assumptions that involve in the computing process of exchange–correlation energy, including Hatree–Fock self-consistent method, Born–Oppenheimer assumption, and the Jacob’s ladder of density functional approximations. Hatree–Fock method assumes that the electrons move independently and repulse each other based on the average positions, and thus can be approximated using a single Slater determinant. Born–Oppenheimer approximation assumes that the lighter electrons adjust adiabatically to the motion of the heavier atomic nuclei, and thus their motions can be separated. Also, depending on how much accuracy one wish to achieve in the results of DFT calculations, there are multiple levels of approximations to the energy–correlation energy, so-called the rungs in Jacob’s ladder, from local spin-density approximation, to generalized gradient approximation (GGA), meta-GGA, hyper-GGA, and generalized random phase approximation [14]. Thus, the potential energy in DFT calculations is numerically approximated rather than exact. Additionally, the PES construction based on the calculated potential energies can be regarded as an interpolation process in a high-dimensional configurational space. Therefore, uncertainty in DFT calculations, which at least includes theoretical and numerical approximations, and numerical interpolation error, is an inevitable element in PES construction.

Efficient methods are needed to reduce the computational complexity in searching on PES. Surrogate modeling or metamodeling is a viable approach, which builds input–output response and numerically approximate the PES. Numerous metamodel methods have been proposed based on a set of sample points, such as polynomial regression [15], support vector regression [16], moving least squares regression [17,18], radial basis functions [19], neural networks [20], kriging method [21], and inverse distance weighting [22]. Strengths and weaknesses exist in different metamodeling approaches. One of the most widely used metamodeling methods is kriging, also known as Gaussian process regression. The kriging metamodel estimator is expressed in terms of polynomial regression, and a Gaussian process with zero mean. Assuming different correlation models between inputs, one can compute the covariance matrix and the unknown polynomial coefficients by ordinary linear regression. Kriging estimator is the best linear unbiased predictor, in the sense that mean-squared error is minimized. The mean-squared error captures the variance of Gaussian process that passes through the provided data points, which is assumed to be independent of the data points’ locations. One of the major challenges for applying kriging is the high computational cost in the sequential sampling and model construction process. If  $m$  is the number of observations, then the computational cost for calculating the inverse of covariance matrix is  $\mathcal{O}(m^3)$ , and the storage cost is  $\mathcal{O}(m^2)$ . Kriging performance decreases as the number of observations  $m$  increases. When the number of data points reaches a thousand, the classical kriging method starts to break down. Several kriging modifications are devised to reduce the computational impact of covariance matrix, as well as to improve the kriging capabilities. Examples include fixed-rank kriging [23], fast kriging algorithm [24], multifidelity co-kriging [25,26], and cluster kriging [27,28]. Lee et al. [29] developed and compared the univariate dimension reduction method, performance moment integration method, and percentile difference methods in reliability-based robust design optimization problem, showing that the dimension reduction method is effective when the number of variables is small, whereas percentile difference method is more effective when the number of random variables is relatively large. Huang and Du [30] proposed the mean-value first-order saddle point approximation to estimate the cumulative and density function of output, linearized by first-order Taylor expansion at the mean values of random input variables. Wang and Wang [31] developed a maximum confidence enhancement-based sequential sampling approach to improve the computational efficiency.

In this paper, a local minimum and saddle point searching method on PES is developed based on symmetry-enhanced cluster kriging metamodels. The objective of the proposed approach is twofold: one is to accelerate the searching process by using kriging predictor instead of relying on DFT functional evaluations, and the other is to improve the performance of classical kriging method for large sample sizes. The searching method begins with a limited number of DFT calculations. When the number of DFT calculations reaches a limit, a new PES metamodel is constructed using the distributed kriging metamodels. Thus, multiple kriging metamodels are constructed as the size of dataset grows. The dataset is divided into clusters dynamically. In each cluster, a classical kriging metamodel is constructed. The PES metamodel predictor is computed as an aggregation of all cluster kriging predictors. The searching method is performed on the PES metamodel until some defined criteria are satisfied. The search results based on the metamodel are used to locate the new configurations for DFT calculations. After DFT calculations are performed on the new configurations, the metamodel is then refined and updated to include those new data points, and the searching method switches back to metamodel. This “real-surrogate-real” process in the searching algorithm continues until some stopping criteria are satisfied. The advantage of the proposed method is the significantly reduced number of functional evaluations using density functional calculation compared to traditional approaches. In addition, the divide-and-reconquer approach with distributed kriging metamodels overcomes the high-dimensionality problems in the configurational space and large sample size. The interpolated information with kriging also comes along with mean-squared error, which is used to assess the uncertainty of the searching method and metamodel.

In the remainder of the paper, Sec. 2 reviews the concurrent searching method developed in our earlier work. Section 3 describes the incorporation of kriging metamodel to accelerate searching method. Section 4 provides the formulation and construction of distributed kriging for high-dimensional and large dataset problem. Section 5 demonstrates the scalability of distributed kriging, in contrast to the computational bottleneck of classical kriging. Section 6 applies the searching method of hydrogen diffusion in body-centered center (BCC) cubic iron using the supporting metamodel to accelerate the searching method. Section 7 concludes the paper.

## 2 Previous Work

We recently developed a concurrent searching method [32] for searching multiple local minima and saddle points simultaneously along one transition pathway on the PES, without the knowledge of the initial and final metastable configurations. The algorithm uses Bézier curve to model the transition path from one local minimum to another. For each polygon formed by the Bézier curve’s control points, two end control points are converged to two local minima, and one intermediate control point climb ups to find the saddle point in between. A constrained degree elevation and reduction scheme is incorporated to maintain an even distribution of control points in modeling Bézier curve. A curve subdivision scheme is developed to break one curve into two curves recursively to possibly search for multiple transition paths. The searching method is composed of three stages: (1) a single transition pathway search, (2) multiple transition pathway search, and (3) climbing process to locate the saddle position.

In the first stage, the algorithm locates two local minima by minimizing two end control points using the conjugate gradient method. The intermediate control points are relaxed along their corresponding conjugate directions with positive eigenvalues of the Hessian matrix. In the second stage, the algorithm locates all local minima between two stable configurations obtained from the first stage. To locate all local minima along the path, a curve subdivision scheme is developed to check whether one curve crosses an extra energy basin with a local minimum between the two

stable states (i.e., end control points of the curve produced from the first stage). If one curve is breakable, the algorithm breaks the curve successively into two curve sections representing two stages of transition, separating at one break point. The break point is then relaxed again using the conjugate gradient method to locate the extra local minimum, while the shape of the two newly created curves are refined in the same manner as in the first stage. This process continues until each curve crosses only two adjacent energy basins. In the third stage, the control point with the maximum energy within each of those transition paths climbs up to locate the actual saddle points. Similar to the first stage, a set of conjugate directions are constructed for each intermediate control point. The control point with the maximum energy along each transition pathway is maximized along the direction with a negative eigenvalue and relaxed along all other conjugate directions. All the other intermediate control points are only relaxed in all the conjugate directions except the one with negative eigenvalues. The algorithm was not implemented to execute in parallel. However, the optimization processes of these curve sections are independent and can be further implemented as parallel searching methods. A number of numerical examples, such as Rastrigin function, Schwefel function, and London–Eyring–Polanyi–Sato potential, are provided to benchmark the searching method [32–34].

### 3 Saddle Point Searching Method Using Potential Energy Surface Metamodel

To improve the efficiency, the searching method is accelerated using a PES metamodel in this work. In this section, we discuss the incorporation of kriging into the single transition pathway search and climbing process using PES metamodel. The incorporation of the PES metamodel is enabled by implementing several thresholds. For instance, if the number of certain functional evaluations passes these thresholds, then the functional evaluations are performed using the PES metamodel instead of DFT calculations, to reduce the computational cost of the searching method. At certain point, the searching method switches back to DFT calculations to update the PES metamodel. This approach is referred to as the “real-surrogate-real” process, by accelerating the algorithm using the PES metamodel, and updating it as the algorithm advances to improve the metamodel accuracy.

Figure 1 summarizes the general process of the searching method using PES metamodels. In the first stage, the algorithm updates the positions of the end control points based on DFT calculations using the conjugate gradient method, and the positions of intermediate control points by moving them along conjugate directions until a number of functional evaluations equals to a threshold. The PES metamodel is constructed, and the searching method relies on sampling scheme to locate two local minima for two end control points. For each end control point, the algorithm draws uniformly distributed samples in a local region, defined by the current location of the end control point, and a hypercube whose side length is

$$a = \sqrt{2}c \min(d_{\text{pre}}, d_{\text{neighbor}}) \quad (1)$$

where  $c \in (0, 1]$  is a constant,  $d_{\text{pre}} = \|\mathbf{x}^{(i)} - \mathbf{x}^{(i-1)}\|$  is the distance between the positions of the end point in the current iteration  $i$  and previous iteration  $(i - 1)$ .  $d_{\text{neighbor}} = \|\mathbf{x}_{\text{end}}^{(i)} - \mathbf{x}^{(i)}\|$  is the distance between the end point and its neighboring control point. The definition of the hypercube length assures that the new end control point will not jump to the position which is far away from the closest local minimum, and prevents the formation of possible loops at the end of the curve. In the second stage, the functional evaluations in the line search along conjugate directions are also based on DFT calculations until another threshold is enabled. When more functional evaluations are required, the PES metamodel is used as an approximation. Line search is one of the main

components in the searching method, as it involves in all stages, and is used to determine the minimum or maximum along each conjugate direction. The line search process conducts a few trial searches to determine an appropriate step length using DFT calculations, refines the position of the control points along the direction using the determined step length also using DFT calculations, and updates the position of control points using the PES metamodel until stopping criteria is satisfied. In the third stage, the control points with the maximum energy climb up along a conjugate direction and all other intermediate control points are minimized along their corresponding conjugate directions with positive eigenvalues. The construction of the conjugate directions is based on DFT calculations, except for some functional evaluations during the line search are conducted using the PES metamodel until a threshold is satisfied. After that, the climbing process depends solely on the PES metamodel.

### 4 Symmetry-Enhanced Distributed Kriging Metamodel

In this section, we describe the symmetry-enhanced distributed kriging. The new kriging approach is a natural extension of classical kriging to (1) take advantages of symmetry in materials systems and (2) break the large datasets into smaller clusters of fixed size and build a kriging model on each cluster.

**4.1 Symmetry in Materials Systems.** In a material system, an atom is invariant to its own kind. The physical interpretation of this statement can vary from one nanoscale computational method to another. Here, we discuss the physical meaning in terms of molecular dynamics simulation and DFT calculations. For molecular dynamics simulation, simply exchanging the atomistic positions and velocities of two atoms of the same kind simultaneously does not affect the dynamics behavior of the simulation system. For DFT calculations, the Born–Oppenheimer approximation assumes that the motion of nuclei and electrons are separable. The symmetry principle is even more useful for ground-state DFT calculations, where atomistic velocities can be disregarded corresponding to 0K temperature. Thus, exchanging the atomistic positions of two atoms of the same kind also does not affect the calculated potential energy. The statement can also be understood

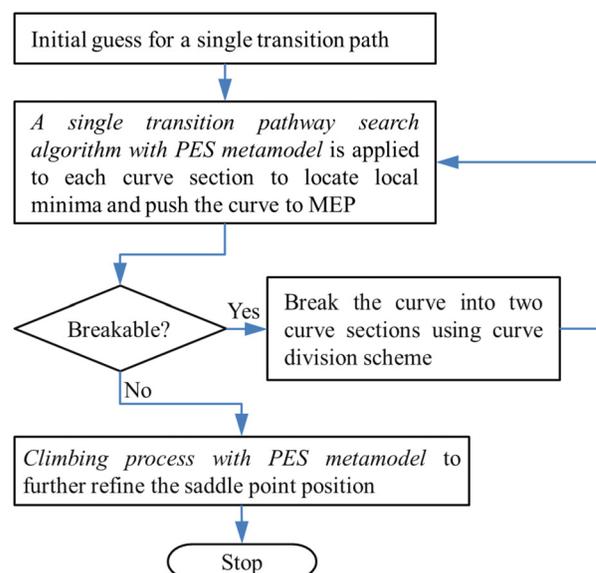


Fig. 1 Flowchart of the searching method with metamodel involves three stages. The first and second stages are combined into a single transition pathway search, and third stage is the climbing process. A PES metamodel is utilized in all three stages to accelerate the searching process.

as the invariant properties of enumeration or relabeling of atoms. In other words, the potential energy is invariant to the permutation of the configurations. Assume that the chemical composition of the material system is  $A_a B_b \dots Z_z$ , and denote the superscript as the enumeration of each component. To illustrate the argument for DFT calculations, Fig. 2 illustrates the concept of swapping atomistic positions without alternating potential energy. For instance, simultaneous exchange of the atomistic position  $A^{(1)} \rightarrow A^{(2)}$ ,  $A^{(2)} \rightarrow A^{(a)}$ , ...,  $A^{(a)} \rightarrow A^{(1)}$ , ...,  $Z^{(1)} \rightarrow Z^{(1)}$ ,  $Z^{(2)} \rightarrow Z^{(z)}$ , ...,  $Z^{(z)} \rightarrow Z^{(2)}$  does not affect the potential energy as illustrated in Fig. 2(a). For the aforementioned chemical composition, the number of permutation is  $a!b!\dots z!$ , meaning that for one data point in DFT calculations, there are  $a!b!\dots z!$  other inputs producing the same output. We refer to other permuted inputs as the symmetric configurations with respect to the real configuration of DFT calculations. The permutation principle is used to produce many symmetric configurations that yield the same potential energy, which in turn increases the number of observations for kriging metamodel and overcomes the curse of dimensionality. One could also use permutation to increase the resolution of kriging, but it is also worthy to note that the metamodel efficiency always reduces as more data points are introduced, especially for very large permutations. It is noteworthy that the PES metamodels are constructed with the permuted configurations, instead of the original configurations.

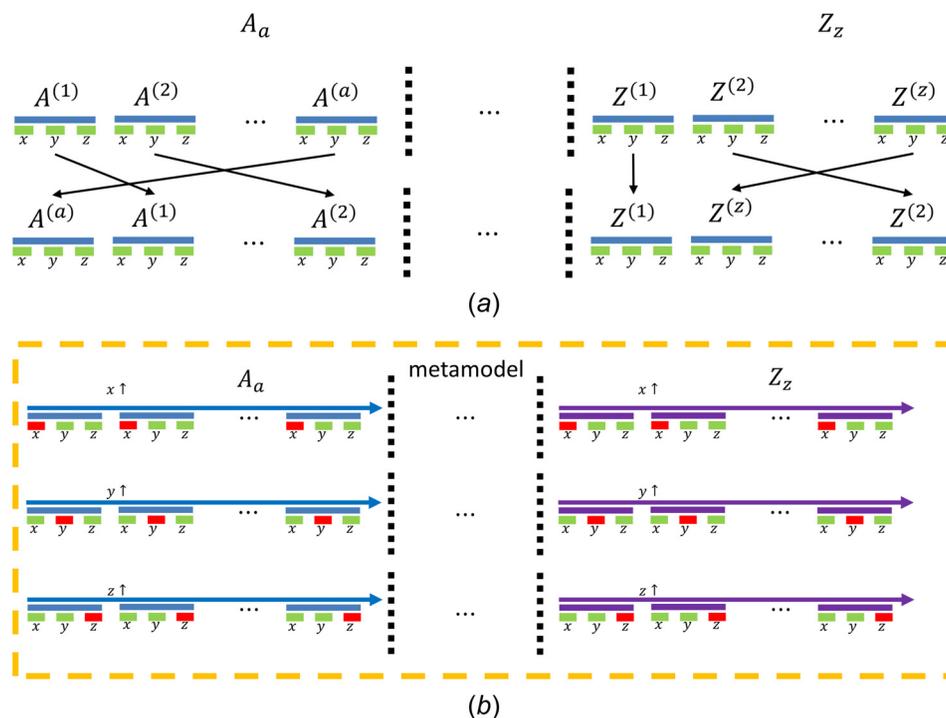
It is shown that scattered data, which are DFT inputs and outputs for PES metamodels, tend to reduce the accuracy of the PES metamodel prediction. Driven by the symmetry principle, DFT inputs and outputs are first permuted in such a way that their distances, typically measured by the Euclidean  $\ell_2$  norm, are minimized, and thus the PES metamodel prediction can be improved. We propose two steps to construct the PES metamodel for configurational space. First, by the aforementioned symmetry principle, the input of DFT calculation, which describes the atomistic

positions of the configuration, is permuted to create other three fictitious DFT inputs. These three DFT inputs are created by sorting the original DFT input so that the  $x$ -,  $y$ -, and  $z$ -coordinates of the original DFT input, respectively, are monotonically increasing for each chemical component of the configuration. Figure 2(b) illustrates the sorting process to these three fictitious DFT inputs. Then, the kriging metamodel is constructed using only these sorted configurations, but not the original DFT inputs. The output of the original DFT calculation is preserved during the sorting process. To predict the functional evaluation for an unknown location using the PES metamodels, the inputs of DFT calculations are first sorted into three different configurations. The predictions of all configurations are evaluated. Among three functional evaluations, the one with the minimum mean-squared error is picked. The advantages of the sort function are twofold. First, the distances between inputs are minimized; thus, the accuracy of the kriging metamodel significantly increases. Second, by carefully selecting three specific permutations of all possible permutations by the symmetry principle, the PES sample size is kept at a computationally tractable level, and the efficiency of the proposed PES metamodels is maintained. The minimization of distances between inputs by sorting is established based on the rearrangement inequality theorem. The theorem states that by sorting, the  $\ell_2$ -norm between two vectors is minimized.

**THEOREM 1. (Rearrangement inequality).** Let  $X, Y$  be the sorted configuration of  $\mathbf{x}, \mathbf{y}$ , respectively, i.e., [35]  $\{x_i\}_{i=1}^n = \{X_i\}_{i=1}^n$ ,  $\{y_i\}_{i=1}^n = \{Y_i\}_{i=1}^n$  and

$$X_i \leq X_j \quad \text{and} \quad Y_i \leq Y_j \quad \text{for every } i, j \quad 1 \leq i \leq j \leq n$$

then  $\|X - Y\|_2^2 = \arg \min_{\sigma(x,y)} \|\mathbf{x} - \mathbf{y}\|_2^2$ , where  $\sigma(x, y)$  is the set of all possible permutations.



**Fig. 2** The invariant property of PES with respect to the atomistic component and the construction of symmetry-enhanced cluster kriging metamodel: (a) The potential energy of an arbitrary material systems is invariant to the permutation of its atomistic component and (b) The construction of symmetry-enhanced cluster kriging metamodel by sorting each of its component by  $x$ -,  $y$ -, and  $z$ -coordinates separately and construct the PES metamodel based on the permuted configurations

## 4.2 Clustering Algorithm.

**Algorithm 1** Sequential construction of clusters given  $k$  initial clusters.

**Input:** new data point  $\{x_{\text{new}}, y_{\text{new}}\}$

- 1: **if**  $\text{cluster}(k).N_{\text{data}} \leq N_{\text{thres}}$  **then**
- 2:   Add new data point into the last cluster:
- 3:    $\text{cluster}(k).X \leftarrow [x_{\text{new}}; \text{cluster}(k).X]$
- 4:    $\text{cluster}(k).y \leftarrow [y_{\text{new}}; \text{cluster}(k).y]$
- 5:   Remove last datapoint from  $\text{cluster}(k).X$  and  $\text{cluster}(k).y$
- 6:   Increase the counter:  $\text{cluster}(k).N_{\text{data}} \leftarrow \text{cluster}(k).N_{\text{data}} + 1$
- 7:   **if**  $N_{\text{fit}} | \text{cluster}(k).N_{\text{data}}$  **then**
- 8:     Fit the hyperparameter  $\theta$  every  $N_{\text{fit}}$  of new datapoints
- 9:   **end if**
- 10: **else**
- 11:   Create  $(k + 1)$  cluster:
- 12:    $\text{cluster}(k + 1).y \leftarrow \text{cluster}(k).y$
- 13:    $\text{cluster}(k + 1).X \leftarrow \text{cluster}(k).X$
- 14:   Add new data point into the last cluster:
- 15:    $\text{cluster}(k).X \leftarrow [x_{\text{new}}; \text{cluster}(k).X]$
- 16:    $\text{cluster}(k).y \leftarrow [y_{\text{new}}; \text{cluster}(k).y]$
- 17:   Remove last datapoint from  $\text{cluster}(k).X$  and  $\text{cluster}(k).y$
- 18:   Reset the counter:  $\text{cluster}(k + 1).N_{\text{data}} \leftarrow 1$
- 19:   Increase the number of cluster:  $k \leftarrow k + 1$
- 20: **end if**

The searching method only switches to the “real-surrogate-real” approach if there is sufficient data to support the metamodel. The initial dataset is first sorted according to the symmetry property as described previously, before being clustered again to minimize their overlapping data. As the searching method advances, more data points are added into the metamodel prediction. Here, we propose a feedforward scheme to build clusters sequentially. The algorithm is inspired by the local Gaussian process regression [27,28]. Suppose we have  $k$  clusters. In each cluster, there is a counter of unique data points  $N_{\text{data}}$ . If the counter exceeds the predefined threshold for the cluster size  $N_{\text{thres}}$ , then the new cluster  $k + 1$  is created in the following manner. First, the counter  $N_{\text{data}}$  for the new cluster is reset to one. Second, the remaining missing data ( $N_{\text{thres}} - N_{\text{data}}$ ) is borrowed from the last cluster  $k$ . Finally, when a new data point is added, the borrowed data point is replaced with the new one until the counter  $N_{\text{data}}$  hits the threshold  $N_{\text{thres}}$ , and a new cluster is created again. As previously mentioned, on each cluster, a kriging model is built. After certain number of steps  $N_{\text{fit}}$ , on each cluster, the hyperparameters  $\theta$  are recalculated by optimizing the maximum likelihood estimation function, and the centroids  $c_k^i$ s are updated. The construction scheme is summarized in Algorithm 1.

## 4.3 Prediction Using Multiple Metamodels.

**Algorithm 2** Prediction using weighted kriging from all clusters.

**Input:** location  $x$ ,  $k$  clusters and their centroids  $c_l, 1 \leq l \leq k$

**Output:** the prediction  $\hat{y}$  at location  $x$  using  $k$  clusters

- 1: **for**  $l \leftarrow 1, k$  **do**
- 2:   Compute the distance from  $x$  to all clusters’ centroids:
- 3:    $d_l^2 \leftarrow \|x - c_l\|_2^2$
- 4:   **if**  $\min_l \{d_l\}_{l=1}^k \neq 0$  **then**
- 5:     Compute the weights with a radial basis function  $f_{\text{RBF}}(\cdot)$ :
- 6:      $w_l \leftarrow f_{\text{RBF}}(d_l)$
- 7:   **else**
- 8:     Insert a penalized term  $d_{\text{avg}}$  to stabilize  $f_{\text{RBF}}(\cdot)$
- 9:     Compute the weights:
- 10:      $w_l \leftarrow f_{\text{RBF}}(d_{\text{avg}}, d_l)$
- 11:   **end if**
- 12: **end for**

13: Normalize the weights:

$$14: \quad w_l \leftarrow \frac{w_l}{\sum_{l=1}^k w_l}$$

15: Compute the prediction:

$$16: \quad \hat{y} \leftarrow \sum_{l=1}^k w_l \bar{y}^{(l)} \triangleright \bar{y}^{(l)} \text{ is the kriging prediction of } l \text{ cluster at } x$$

To predict the response value  $\hat{y}$  at the location  $x$ , given  $k$  clusters and a kriging metamodel on each cluster, the prediction is given in the form of weighted average as

$$\hat{y} = \sum_{l=1}^k w_l \bar{y}_l \quad (2)$$

where  $\sum_{l=1}^k w_l = 1$ , and  $\bar{y}_l$  is the kriging prediction of the  $l$ th cluster. Here, the local weight  $w_l$  is computed based on a radial basis function  $f_{\text{RBF}}(\cdot)$  of the distance  $d$  from the query point  $x$  to the centroid of the  $l$ th cluster  $c_l$ , i.e.,  $d = |x - c_l|$ , for example, inverse distance weight  $w_l \propto (1/d_l^2)$ , or Gaussian weight  $w_l \propto e^{-\alpha d_l^2}$ . The main principle for choosing an appropriate weight function is that the closer the location to the centroid of the  $l$ th cluster, the more accurate prediction one can obtain from this kriging prediction. As the searching method advances, the location point  $x$  for prediction tends to be away from initial datasets. With the formalism of sequential clusters construction, only the last few clusters contribute meaningful predictions to the location  $x$ . In case the prediction is at the centroid of one or many clusters, a penalized term is inserted to the denominator to avoid numerical instabilities. Nguyen et al. [28] proposed the weights computed by a Gaussian kernel of the distance to each clusters. van Stein et al. [27] minimized the variance of the weighted average and obtained the weights of inverse variances. More general choices for weight functions exist in literature, such as Matérn functions or the multi-quadrics [36] with generalized exponentials. The prediction algorithm is summarized in Algorithm 2. Because the underlying assumption of kriging is that the errors are Gaussian, the predicted variance can be assessed as

$$\hat{\sigma}^2 = \sum_{l=1}^k w_l^2 \bar{\sigma}_l^2 \quad (3)$$

where  $\bar{\sigma}_l^2$  is the predicted mean-squared error from the kriging model of the  $l$ th cluster, assuming data in clusters are independently sampled.

## 5 Numerical Validation and Verification Example

Here, we demonstrate the scalability of the distributed kriging with the Schwefel function on two-dimensional, where the function is evaluated by

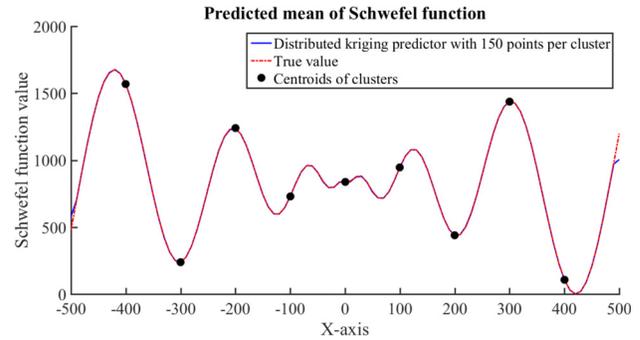
$$f(x) = 418.9829 \times 2 - (x \sin(\sqrt{|x|}) + y \sin(\sqrt{|y|})) \quad (4)$$

on the domain  $[-500, 500] \times [-500, 500]$ . In order to compare the classical and distributed kriging, the subdomain of diagonal stripe is chosen where nine clusters of distributed kriging are constructed. This comparison subdomain is the union of nine squares, where each square is associated with a cluster in distributed kriging. The centers of the squares are  $(-400, -400)$ ,  $(-300, -300)$ ,  $\dots$ ,  $(400, 400)$ . The dimension of each square is 200. In this example, the Gaussian weight function is chosen, in which the weights are exponentially decayed with respect to  $d_l$ , i.e.,  $w_l \propto e^{-0.01 d_l}$ , where  $d_l$  is the distance between query point to the  $l$ th cluster’s centroid. Figures 3(a) and 3(b) shows the interpolatory surfaces between classical and distributed kriging, respectively, on the comparison domain of diagonal stripe, where both interpolatory surfaces are very similar.

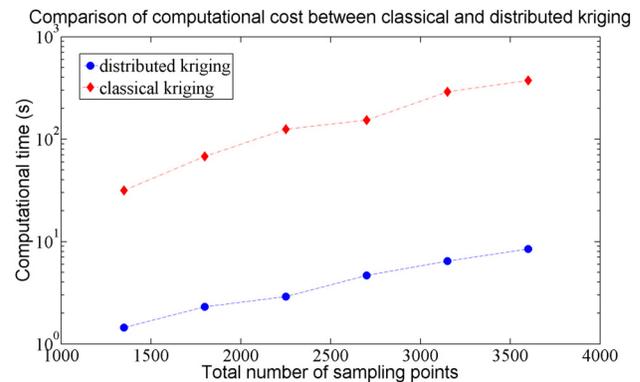
To verify the computation, the distributed kriging is compared against with true Schwefel function value with 101 equidistant points on the line  $x = y$  in  $[-500, 500] \times [-500, 500]$ , where the cluster size  $N_{\text{data}}$  is set to 150. The comparison results between the distributed kriging and the true value of Schwefel function are presented in Fig. 4, where the Schwefel function value is plotted against the  $x$ -coordinates, showing an excellent agreement. Note that near the boundary of the computational domain, the distributed kriging predictor slightly deviates from the true function due to less number of available sampling points around the neighborhood. The clusters' centroids are plotted as a black dot along the curve.

To show the scalability of distributed kriging, we compare the computational cost between distributed and classical kriging, where the total number of sampling points varies at 1350, 1800, 2250, 2700, 3150, and 3600. These numbers correspond to the cluster size  $N_{\text{data}}$  of 150, 200, 250, 300, 350, and 400. Figure 5 presents the comparison of computational cost between classical and distributed kriging. With  $N_{\text{data}} = 400$ , corresponding to the total number of sampling points of 3600, the classical kriging takes 376.31 s to construct the metamodel, whereas the distributed kriging only takes 8.50 s. It is demonstrated that the scalability problem of classical kriging can be solved by decomposing large dataset into smaller clusters, where each clusters corresponds to a kriging metamodel. The advantage of distributed kriging is that the response surface can be constructed at a much computationally cheaper price.

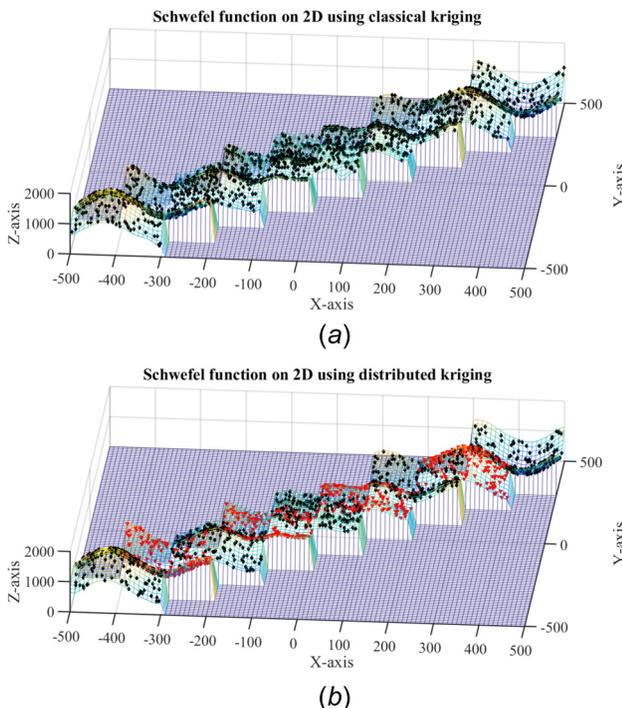
Theoretically, the distributed kriging method in optimization would yield the same result compared to the classical kriging, provided that these constraints are satisfied. First, there must be sufficient number of data points in each cluster to approximate the local response surface. Second, the clusters have to be relatively disjoint from others to keep the clusters' centroids substantially



**Fig. 4 Comparison between the distributed kriging and the true value of Schwefel function shows an excellent agreement with a small number of cluster size  $N_{\text{data}}$  of 150**



**Fig. 5 Comparison of computational cost shows that the distributed kriging scheme is scalable**

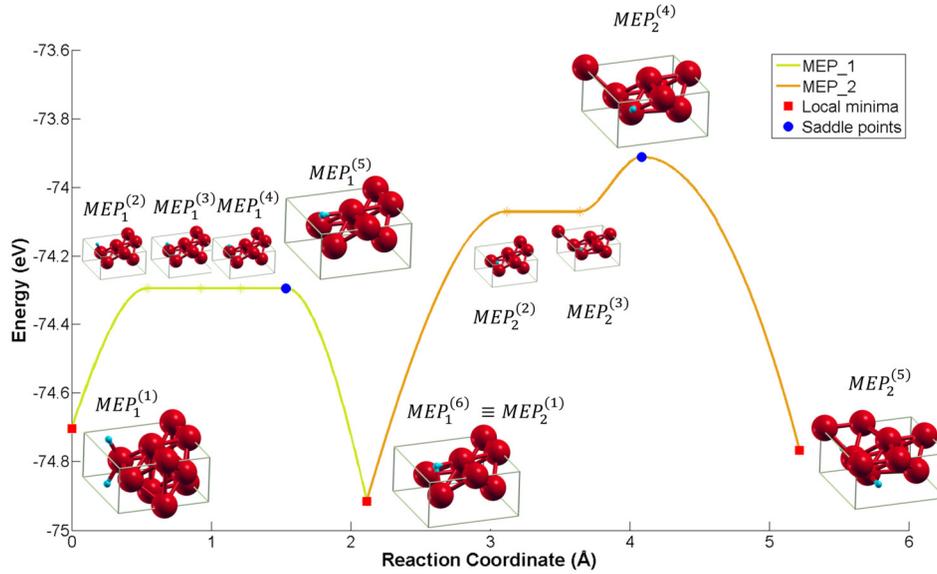


**Fig. 3 Comparison between classical and distributed kriging on the comparison domains shows very similar response surfaces (a) Classical kriging on the comparison domain with 1800 sampling points and (b) Distributed kriging on the comparison domain, where the sampling points on odd clusters are plotted as diamonds, and the sampling points on even clusters are plotted as upside down triangle. There are totally nine clusters, where each cluster contains 200 sampling points.**

far away from each other. Third, there must also be some overlapping regions between clusters, so that the prediction between clusters is accurate. Finally, an appropriate weight computation needs to be carefully chosen. In practice, the optimization results might vary due to unsatisfied constraints. The distributed kriging uncertainty is typically higher compared to classical kriging, due to significantly less number of data points available in the cluster. Therefore, it is recommended to set the cluster size sufficiently high, but not too high, preferably in the magnitude of  $10^3$ . Furthermore, the buffering zone that connects one cluster to another should also have sufficient sampling points to prevent spikes of mean-squared errors. Locally, the accuracy of distributed kriging response surface is essentially the same with classical kriging, i.e., the approximation error depends on the number of sampling points. The more sampling points there are, the smaller the approximation error is.

## 6 Material Science Example: Hydrogen Embrittlement in Body-Centered Cubic Iron

An example of hydrogen embrittlement in BCC iron, in which the hydrogen atom diffuses in pure BCC iron with lattice constant of 2.86 Å, is used to demonstrate the searching method. There are two possible sites that H atoms reside in BCC iron: one is octahedral site and another is tetrahedral site. Since it is generally believed that hydrogen has low solubility in BCC iron, we assume there is one hydrogen in the supercell, which is composed of four unit cells and includes eight Fe atoms. The chemical composition of the material system is  $\text{Fe}_8\text{H}$ . The lattice parameter for the unit cell  $\text{Fe}_8\text{H}$  is set at  $a = 5.72$  Å,  $b = 2.86$  Å, and  $c = 5.72$  Å. The total energy of the system and the forces on each atom are performed based on DFT calculations using Vienna Ab initio simulation



**Fig. 6** Transition pathway with local minima and saddle points from initial to final states and its corresponding configurations for  $\text{Fe}_8\text{H}$  system. The saddle points are denoted as circles, whereas the local minima are denoted as squares.

package. The local density approximation projector augmented wave potential is used. The convergence test for the  $k$ -point sampling with respect to the total energy shows that  $13 \times 26 \times 13$  gamma centered grid of  $k$ -point sampling is adequate for study of  $\text{Fe}_8\text{H}$  structure. To reduce the computational time,  $k$ -point sampling of  $2 \times 4 \times 2$  is used. In this simulation, the cluster size  $N_{\text{data}}$  is set at 721. The cluster size is chosen so that it is less than 1000, but very close to the least common multiple of 2, 3, 4, 5, and 6 so that the  $N_{\text{fit}}$  parameter for fitting procedure can be easily chosen from these numbers. The number of DFT calculations is 1598. For every  $N_{\text{fit}}=4$  steps, the hyperparameters in the last cluster are updated. In our study, the variances of clusters are roughly of the same scale, and the exponential weights decay rapidly to zero without the shape parameters in the exponent, leading to numerical instability issues. Thus, the inverse distance weight function is used to compute the weight, i.e.,  $w_l = (1/d_l^2)$ , where  $d_l$  is the distance from the query point to the  $l$ th cluster. The penalized parameter  $d_{\text{avg}}^2$  is set at  $0.25 \cdot (\max_l \{d_l\}_{l=1}^k)^2$ . The uncertainty of the searching method, particularly the predicted energy levels at local minima and saddle points, and the MEP are assessed based on Eq. (3). The PES metamodels helps to predict not only the potential energy from DFT calculations, but also its first partial derivatives with respect to each of the atom positions.

Figure 6 shows the transition pathways obtained by the proposed searching method using the PES metamodel. The corresponding configurations at different states are also presented corresponding with the local minima and saddle points of the transition path. Table 1 presents the potential energy of the local minima, saddle points, and control points along the curve of MEP<sub>1</sub> and MEP<sub>2</sub>. The activation energy is calculated to be 1.004 eV. The computational time is about 31 h, using four processors for

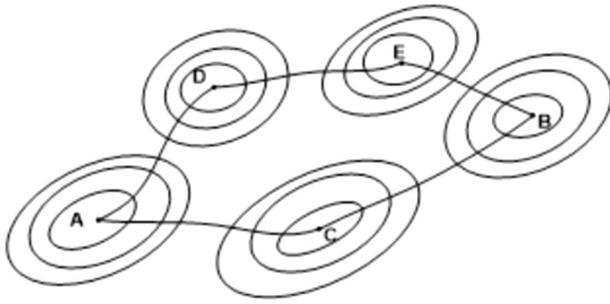
parallel DFT calculations. The covariance matrix size is  $721 \times 721$  for each of the two clusters.

The proposed distributed kriging breaks the covariance matrix in classical kriging method into several covariance matrices, one for each kriging metamodel on each cluster. The covariance matrix has a chosen constant size. By doing so, the fitting hyperparameters can be calculated at a cheaper computational price. Additionally, only the hyperparameters for the kriging metamodel for the last cluster are inferred. Thus, the nature of sequential design is preserved in the distributed kriging approach. Furthermore, because the size of covariance matrices is smaller than the classical kriging approach, the distributed kriging is more efficient. A comparison between classical kriging and distributed kriging depends on many parameters, but mainly on the chosen size of the cluster  $N_{\text{data}}$  and the number of clusters  $k$ . If  $m$  is the sample size, the computational cost reduces from  $\mathcal{O}(m^3)$  for classical kriging to  $\mathcal{O}(kN_{\text{data}}^3)$  for distributed kriging, almost by a factor of  $k^2$ , where  $m \leq kN_{\text{data}}$ . In this example, the transition path of the hydrogen atom is known a priori that the metastable and final states are away from the initial one. With the sequential formalism of cluster construction, as the searching algorithm converges, we expect clusters are born naturally along the pathway in a sequential manner. Therefore, the data are automatically clustered and no clustering algorithm is needed. For a more general application, one may need to consider clustering algorithms, such as  $k$ -means and random clustering, to cluster the large dataset after a certain number of clusters are constructed.

While the idea of using PES metamodels to support search method is promising in term of computational time and the simple reconstruction of potential energy landscape, there are several limitations in the current state of the search method. First, even

**Table 1** Potential energy of the configurations in Fig. 6 and their standard deviations

		Configuration number					
		1	2	3	4	5	6
Pot. E.	MEP <sub>1</sub>	-74.7027	-74.2947	-74.2945	-74.2943	-74.2942	-74.9158
	Std. Dev.	3.2652	3.2825	3.2951	13.5816	13.5033	3.3306
	MEP <sub>2</sub>	-74.9158	-74.0713	-74.0712	-73.9122	-74.7660	
	Std. Dev.	3.3306	13.4424	13.3451	13.2758	12.9051	



**Fig. 7 Illustration for two possible transition paths between two states**

though the distributed kriging can compute the functional evaluations quickly compared to the classical kriging method, the uncertainty associated with distributed kriging is typically higher because distributed kriging has significantly fewer data points, hence a larger standard deviation. Additionally, the uncertainty is amplified in a high-dimensional configurational space. However, the location of saddle points is assured by the gradients of PES, whose gradient vectors contain mostly zero components. Second, the current curve subdivision scheme that breaks the modeling Bézier curves successively is incapable of searching two paths with the same initial and final states. There could be multiple paths between two states, as illustrated in Fig. 7.

The current search method can only locate one transition path which consists of multiple curves with their end points connected together locating at multiple local minima, that is A-D-E-B or A-C-B, but not both. As a result, the activation energy between states A and B can be overestimated.

## 7 Conclusion

In this paper, a saddle point searching method using distributed kriging metamodels is developed to improve the efficiency of the searching method. Furthermore, the distributed kriging metamodels are refined to cope with the large sample size and high-dimensionality problems. The main improvement from existing saddle point search methods is that the new searching method retains the searching history in the PES metamodels. These metamodels are continuously updated as the algorithm advances. Numerical scheme to construct the clusters sequentially is devised, and prediction by inverse distance weighted average is used. Also, the uncertainty of the results is also assessed using the error estimated in kriging metamodels. Possible future directions are stochastic kriging [37], where input uncertainty is included and composite Gaussian process [38], where the global trend and local details are captured separately. The choice of weights in distributed kriging remains an open question for further research.

## Funding Data

- National Science Foundation (Grant Nos. CMMI-1001040 and CMMI-1306996).

## References

- [1] Quapp, W., and Heidrich, D., 1984, "Analysis of the Concept of Minimum Energy Path on the Potential Energy Surface of Chemically Reacting Systems," *Theor. Chim. Acta*, **66**(3–4), pp. 245–260.
- [2] Berne, B. J., Ciccolini, G., and Coker, D. F., 1998, *Classical and Quantum Dynamics in Condensed Phase Simulations*, World Scientific, Singapore.
- [3] Laidler, K. J., and King, M. C., 1983, "Development of Transition-State Theory," *J. Phys. Chem.*, **87**(15), pp. 2657–2664.
- [4] Henkelman, G., Jónsson, H., and Jónsson, H., 2002, "Methods for Finding Saddle Points and Minimum Energy Paths," *Theoretical Methods in Condensed Phase Chemistry*, Springer, Dordrecht, The Netherlands, pp. 269–302.
- [5] Schlegel, H. B., 2003, "Exploring Potential Energy Surfaces for Chemical Reactions: An Overview of Some Practical Methods," *J. Comput. Chem.*, **24**(12), pp. 1514–1527.

- [6] Alhat, D., Lasrado, V., and Wang, Y., 2008, "A Review of Recent Phase Transition Simulation Methods: Saddle Point Search," *ASME Paper No. DETC2008-49411*.
- [7] Lasrado, V., Alhat, D., and Wang, Y., 2008, "A Review of Recent Phase Transition Simulation Methods: Transition Path Search," *ASME Paper No. DETC2008-49410*.
- [8] Jónsson, H., Mills, G., and Jacobsen, K. W., 1998, "Nudged Elastic Band Method for Finding Minimum Energy Paths of Transitions," *Classical and Quantum Dynamics in Condensed Phase Simulations*, Vol. 385, World Scientific, Singapore.
- [9] Henkelman, G., and Jónsson, H., 2000, "Improved Tangent Estimate in the Nudged Elastic Band Method for Finding Minimum Energy Paths and Saddle Points," *J. Chem. Phys.*, **113**(22), pp. 9978–9985.
- [10] Ionova, I. V., and Carter, E. A., 1993, "Ridge Method for Finding Saddle Points on Potential Energy Surfaces," *J. Chem. Phys.*, **98**(8), pp. 6377–6386.
- [11] Dewar, M. J., Healy, E. F., and Stewart, J. J., 1984, "Location of Transition States in Reaction Mechanisms," *J. Chem. Soc., Faraday Trans. 2*, **80**(3), pp. 227–233.
- [12] Henkelman, G., and Jónsson, H., 1999, "A Dimer Method for Finding Saddle Points on High Dimensional Potential Surfaces Using Only First Derivatives," *J. Chem. Phys.*, **111**(15), pp. 7010–7022.
- [13] Simons, J., Joergensen, P., Taylor, H., and Ozment, J., 1983, "Walking on Potential Energy Surfaces," *J. Phys. Chem.*, **87**(15), pp. 2745–2753.
- [14] Perdew, J. P., Ruzsinszky, A., Constantin, L. A., Sun, J., and Csonka, G. I., 2009, "Some Fundamental Issues in Ground-State Density Functional Theory: A Guide for the Perplexed," *J. Chem. Theory Comput.*, **5**(4), pp. 902–908.
- [15] Queipo, N. V., Haftka, R. T., Shyy, W., Goel, T., Vaidyanathan, R., and Tucker, P. K., 2005, "Surrogate-Based Analysis and Optimization," *Prog. Aerosp. Sci.*, **41**(1), pp. 1–28.
- [16] Clarke, S. M., Griebisch, J. H., and Simpson, T. W., 2005, "Analysis of Support Vector Regression for Approximation of Complex Engineering Analyses," *ASME J. Mech. Des.*, **127**(6), pp. 1077–1087.
- [17] Levin, D., 1998, "The Approximation Power of Moving Least-Squares," *Math. Comput. Am. Math. Soc.*, **67**(224), pp. 1517–1531.
- [18] Kim, C., Wang, S., and Choi, K. K., 2005, "Efficient Response Surface Modeling by Using Moving Least-Squares Method and Sensitivity," *AIAA J.*, **43**(11), pp. 2404–2411.
- [19] Keane, A., 2004, "Design Search and Optimisation Using Radial Basis Functions With Regression Capabilities," *Adaptive Computing in Design and Manufacturing VI*, Springer, London, pp. 39–49.
- [20] Haykin, S., 2004, *Neural Networks: A Comprehensive Foundation*, 2nd ed., Prentice Hall, Upper Saddle River, NJ.
- [21] Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P., 1989, "Design and Analysis of Computer Experiments," *Statist. Sci.*, **4**(4), pp. 409–423.
- [22] Lu, G. Y., and Wong, D. W., 2008, "An Adaptive Inverse-Distance Weighting Spatial Interpolation Technique," *Comput. Geosci.*, **34**(9), pp. 1044–1055.
- [23] Cressie, N., and Johannesson, G., 2008, "Fixed Rank Kriging for Very Large Spatial Data Sets," *J. R. Stat. Soc. Ser. B*, **70**(1), pp. 209–226.
- [24] Sakata, S., Ashida, F., and Zako, M., 2004, "An Efficient Algorithm for Kriging Approximation and Optimization With Large-Scale Sampling Data," *Comput. Methods Appl. Mech. Eng.*, **193**(3), pp. 385–404.
- [25] Myers, D. E., 1982, "Matrix Formulation of Co-Kriging," *J. Int. Assoc. Math. Geol.*, **14**(3), pp. 249–257.
- [26] Forrester, A. I., Sobester, A., and Keane, A. J., 2007, "Multi-Fidelity Optimization Via Surrogate Modelling," *Proc. R. Soc. London A*, **463**(2088), pp. 3251–3269.
- [27] van Stein, B., Wang, H., Kowalczyk, W., Bäck, T., and Emmerich, M., 2015, "Optimally Weighted Cluster Kriging for Big Data Regression," *International Symposium on Intelligent Data Analysis (IDA)*, Saint-Etienne, France, Oct. 22–24, pp. 310–321.
- [28] Nguyen-Tuong, D., Seeger, M., and Peters, J., 2009, "Model Learning With Local Gaussian Process Regression," *Adv. Rob.*, **23**(15), pp. 2015–2034.
- [29] Lee, I., Choi, K., Du, L., and Gorsich, D., 2008, "Dimension Reduction Method for Reliability-Based Robust Design Optimization," *Comput. Struct.*, **86**(13), pp. 1550–1562.
- [30] Huang, B., and Du, X., 2008, "Probabilistic Uncertainty Analysis by Mean-Value First Order Saddlepoint Approximation," *Reliab. Eng. Syst. Saf.*, **93**(2), pp. 325–336.
- [31] Wang, Z., and Wang, P., 2014, "A Maximum Confidence Enhancement Based Sequential Sampling Scheme for Simulation-Based Design," *ASME J. Mech. Des.*, **136**(2), p. 021006.
- [32] He, L., and Wang, Y., 2013, "A Concurrent Search Algorithm for Multiple Phase Transition Pathways," *ASME Paper No. DETC2013-12362*.
- [33] He, L., and Wang, Y., 2015, "A Curve Swarm Algorithm for Global Search of State Transition Paths," *Third World Congress on Integrated Computational Materials Engineering (ICME)*, Colorado Springs, CO, May 31–June 4, pp. 139–146.
- [34] He, L., 2015, "Multiple Phase Transition Path and Saddle Point Search in Computer Aided Nano Design," *Ph.D. dissertation*, Georgia Institute of Technology, Atlanta, GA.
- [35] Hardy, G. H., Littlewood, J. E., and Pólya, G., 1952, *Inequalities*, Cambridge University Press, Cambridge, UK.
- [36] Fasshauer, G. E., and Zhang, J. G., 2007, "On Choosing 'Optimal' Shape Parameters for RBF Approximation," *Numer. Algorithms*, **45**(1–4), pp. 345–368.
- [37] Ankenman, B., Nelson, B. L., and Staum, J., 2010, "Stochastic Kriging for Simulation Metamodeling," *Oper. Res.*, **58**(2), pp. 371–382.
- [38] Ba, S., and Joseph, V. R., 2012, "Composite Gaussian Process Models for Emulating Expensive Functions," *Ann. Appl. Stat.*, **6**(4), pp. 1838–1860.