Structural and Multidisciplinary Optimization (2020) 61:39–57
https://doi.org/10.1007/s00158-019-02342-3

RESEARCH PAPER

An efficient kriging modeling method for high-dimensional design problems based on maximal information coefficient

Liang Zhao[1] · Peng Wang[1,2] · Baowei Song[1,2] · Xinjing Wang[1] · Huachao Dong[1]

Received: 11 March 2019 / Revised: 22 May 2019 / Accepted: 25 June 2019 / Published online: 24 July 2019

Abstract

Kriging, one of the most popular surrogate models, is widely used in computationally expensive optimization problems to improve the design efficiency. However, due to the "curse-of-dimensionality," the time for generating the kriging model increases exponentially as the dimension of the problem grows. When it comes to the cases that the kriging model needs to be frequently constructed, such as sequential sampling for kriging modeling or global optimization based on kriging model, the increased modeling time should be taken into consideration. To overcome this challenge, we propose a novel kriging modeling method which combines kriging with maximal information coefficient (MIC). Taking the features of the optimized hyper-parameters into consideration, MIC is utilized for estimating the relative magnitude of hyper-parameters. Then this knowledge of hyper-parameters is incorporated into the maximum likelihood estimation problem to reduce the dimensionality. In this way, the high dimensional optimization can be transformed into a one-dimensional optimization, which can significantly improve the modeling efficiency. Five representative numerical examples from 20-D to 80-D and an industrial example with 35 variables are used to show the effectiveness of the proposed method. Results show that compared with the conventional kriging, the modeling time of the proposed method can be ignored, while the loss of accuracy is acceptable. For the problems with more than 40 variables, the proposed method can even obtain a more accurate kriging model with given computational resources. Besides, the proposed method is also compared with KPLS (kriging combined with the partial least squares method), another state-of-the-art kriging modeling method for high-dimensional problems. Results show that the proposed method is more competitive than KPLS, which means the proposed method is an efficient kriging modeling method for high-dimensional problems.

Keywords Kriging · Maximal information coefficient · High-dimensional problems · Metamodels

List of symbols

Matrices and vectors are in bold type

Symbols	Meaning
$\mathbb{R}$	Set of real numbers
$\mathbb{R}^+$	Set of positive real numbers
d	Dimensions
m	Number of sample points
$\mathbf{x}$	A sample point ($1 \times d$ vector)
x_j	jth element of $\mathbf{x}$ for $j = 1, ..., d$
$\mathbf{X}$	$m \times d$ matrix of sample points
$\mathbf{x}^{(i)}$	ith sample point for $i = 1, ..., m$ ($1 \times d$ vector)
$\mathbf{x}_i$	ith column of $\mathbf{X}$ for $i = 1, ..., d$ ($m \times 1$ vector)
$x_l^{(i)}$	ith element of $\mathbf{x_l}$ for $i = 1, ..., m$.
$\mathbf{y}$	$m \times 1$ vector of response values
$y^{(i)}$	Response value of $\mathbf{x}^{(i)}$ for $i = 1, ..., m$
$\hat{y}(\mathbf{x})$	Prediction of a sample point
$z(\mathbf{x})$	Realization of a stochastic process
R	Spatial correlation function
$\mathbf{R}$	Correlation matrix
$\mathbf{1}$	n−vector of ones
$s^2(\mathbf{x})$	Prediction of the kriging variance
σ^2	Process variance
$\boldsymbol{\theta}$	Hyper-parameters ($1 \times d$ vector)
θ_i	ith element of $\boldsymbol{\theta}$ for $i = 1, ..., d$
S_i	First-order Sobol' index for $i = 1, ..., d$
w_i	MIC value of $\mathbf{x_i}$ and $\mathbf{y}$ for $i = 1, ..., d$
λ	Auxiliary parameter

Responsible Editor: Nam Ho Kim

✉ Peng Wang
wangpeng305@nwpu.edu.cn

[1] School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an, China

[2] Key Laboratory of Unmanned Underwater Vehicle Technology, Northwestern Polytechnical University, Xi'an, China

1 Introduction

Over the past two decades, surrogate-based global optimization has played an increasingly important role in many areas of engineering and science (Dong et al. 2019). By constructing surrogate models to approximate or take place of the time-consuming computer simulations, the design efficiency can be impressively improved. Surrogate models are also known as metamodels or response surfaces. There are a number of surrogate models available in the literatures. The representative surrogate models include polynomial response surface model (Schmit and Farshi 1974; Box and Draper 1987), kriging (Krige 1951; Sacks et al. 1989), radial basis function model (Buhmann 2003; Mullur and Messac 2005), and support vector regression (Smola and Schölkopf 2004). Among them, kriging is one of the most popular methods, because it can represent nonlinear and multidimensional functions and has a unique feature of offering a mean-squared-error estimation. Kriging, also known as Gaussian process (Rasmussen and Williams 2005), is a statistical interpolation method suggested by Krige (1951) and mathematically formulated by Matheron (1963). Owing to the research work of Sacks et al. (1989), kriging becomes more and more popular (Dong et al. 2018).

Although the basic theory of kriging has almost been developed for seven decades, it still suffers from some drawbacks for high-dimensional problems. As shown in Liu et al. (2014), constructing a kriging model using 150 points for a 50-D problem with MATLAB optimization toolbox takes from 240 to 400 s. For high-dimensional problems, it seems that building a kriging model itself is a computationally expensive task and even spends more time than running a computer simulation. This drawback limits the application of kriging for high-dimensional optimization problems where the kriging model needs to be frequently constructed. Why is it so time-consuming to build a kriging model for high-dimensional problems? On the one hand, when constructing the kriging model, a key process is estimating the values of the hyper-parameters. This process requires inverting the covariance matrix for several times. The size of covariance matrix is $m \times m$, where m is the number of training points. Loeppky et al. (2009) showed that the appropriate initial sample size for training a kriging model should be ten times the dimensionality. As the dimensionality increases, a larger m is required if we want to obtain a kriging model with great accuracy. As a result, inverting the covariance matrix is computationally expensive. On the other hand, the hyper-parameters are often obtained by maximizing the likelihood function. This is an optimization sub-problem. If the number of design variables is large, the design space of this optimization sub-problem will be a vast space. To find the global optimum, a large amount of computational demand is needed. This kind of difficulty by the dimensionality is known as the "curse-of-dimensionality" (Shan and Wang 2010). Emmerich et al. (2006) pointed out that the time complexity of building a kriging model is $O(N_{it}m^3d)$, where N_{it} is the number of iterations, d is the number of variables. For a high-dimensional problem, a large N_{it} is required due to the vast search space. Therefore, for a high-dimensional problem, it is very difficult to build a high-quality kriging model with given computational effort using traditional modeling methods. Many recent works have addressed this challenge of high-dimensional kriging model (Bouhlel et al. 2016; Hartwig and Bestle 2017; Wang et al. 2017; Lee et al. 2019).

To overcome this challenge, three feasible strategies can be considered: (1) integrate kriging model with high-dimensional model representation (HDMR), (2) reduce the number of training points while maintaining model accuracy by incorporating auxiliary information, and (3) reduce the number of parameters we need to optimize when estimating hyper-parameters. The first two strategies are investigated by many researchers in recent years, such as gradient-enhanced kriging based on HDMR (Ulaganathan et al. 2016a), HDMR using multi-fidelity samples (Cai et al. 2017), weighted gradient-enhanced kriging (Han et al. 2017), and screening-based gradient-enhanced kriging (Chen et al. 2019). The third strategy focuses on reducing the dimensionality of the optimization sub-problem to improve the efficiency for estimating hyper-parameters. A common way is to transform a set of original variables into a smaller set of new variables that retain most of the original information. For example, Liu et al. (2014) applied Sammon mapping technique to transform the design variables to a lower dimensional space. In their work, only four hyper-parameters were optimized when constructing the kriging model for medium-scale problems (20–50 decision variables). Using the partial least squares (PLS) technique, Bouhlel et al. (2016) proposed an effective kriging modeling method named KPLS for high-dimensional problems. KPLS can reduce the number of hyper-parameters to a maximum of 4 parameters and the modeling time can be significantly reduced. To reduce the number of hyper-parameters when building gradient-enhanced kriging, Bouhlel and Martins (2019) proposed a new gradient-enhanced kriging by PLS. In this article, we follow the third strategy and a novel dimension reduction method is proposed to improve the modeling efficiency.

This paper is motivated by the aspiration of developing a novel kriging modeling method which could build a kriging model with a little amount of computational effort for high-

dimensional problems. The proposed method combines kriging with maximal information coefficient (MIC) and is termed as KMIC. MIC is used to estimate the relative magnitude of the optimized hyper-parameters because both the optimized hyper-parameters and MIC can be used for global sensitivity analysis. To reduce the number of parameters that need to be optimized when estimating hyper-parameters, the maximum likelihood estimation problem is reformulated by adding a set of equality constraints. With our approach, only one auxiliary parameter is needed to optimize when estimating hyper-parameters. As a consequence, the modeling efficiency is dramatically improved. The rest of the paper is organized as follows. Section 2 provides a brief introduction of kriging, including the theoretical basis of the kriging, the main steps for constructing kriging, and the relationship between optimized hyper-parameters and global sensitivities. The proposed method is described in detail in Section 3. The performance of KMIC is tested and compared in Section 4. Finally, conclusions are described in Section 5.

2 Kriging

Suppose that we want to build a kriging model for an unknown deterministic function $y = f(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^d$. m sample points $\mathbf{x}^{(1)}, ..., \mathbf{x}^{(m)}$ and the corresponding response values $y^{(1)}, ..., y^{(m)}$ are given. These sample points could be aggregated in a matrix

$$\mathbf{X} = \left[\mathbf{x}^{(1)}, ..., \mathbf{x}^{(m)}\right]^T, \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^{m \times d}$. The corresponding response values can be aggregated as a $m \times 1$ vector

$$\mathbf{y} = \left[y^{(1)}, ..., y^{(m)}\right]^T = \left[f\left(\mathbf{x}^{(1)}\right), ..., f\left(\mathbf{x}^{(m)}\right)\right]^T. \tag{2}$$

2.1 The basic theory of kriging

The basic assumption of kriging model is that the true deterministic response is realized with a trend function and a stochastic process. There are different versions of kriging such as "simple kriging," "ordinary kriging," and "universal kriging." The trend function is the main difference between them (Sasena 2002). Among these different variants, ordinary kriging is the most widely used kriging technique and is used for the experiments in this paper. The proposed method in this research can also be expressed in the same way for the other variants. In ordinary kriging, we assume that the trend function is an unknown constant. The prediction formulation can be written as follows:

$$\hat{y}(\mathbf{x}) = \beta_0 + z(\mathbf{x}), \tag{3}$$

where β_0 is the unknown constant. $z(\mathbf{x})$ is a random process having mean zero and covariance of

$$Cov\left[z\left(\mathbf{x}^{(i)}\right), z\left(\mathbf{x}^{(j)}\right)\right] = \sigma^2 R\left(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}\right) \tag{4}$$

between $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(j)}$, where σ^2 is the process variance and $R(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$ is known as spatial correlation function. Throughout the literature, there are many choices of the spatial correlation function (see Appendix Table 8). In this work, we use the Gaussian exponential correlation function:

$$R\left(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}\right) = \exp\left(-\sum_{k=1}^{d} \theta_k \left|x_k^{(i)} - x_k^{(j)}\right|^2\right), \forall \theta_k \in \mathbb{R}^+, \tag{5}$$

where θ_k is known as hyper-parameter. The number of hyper-parameters here is equal to the number of the design variables. For the sake of brevity, we aggregate the hyper-parameters as a $1 \times d$ vector $\boldsymbol{\theta}$.

Under the hypothesis above, the best linear unbiased predictor for $y(\mathbf{x})$ can be obtained as (Sacks et al. 1989; Jones 2001)

$$\hat{y}(\mathbf{x}) = \beta^* + \mathbf{r}(\mathbf{x})^T \mathbf{R}^{-1}\left(\mathbf{y} - \mathbf{1}\beta^*\right), \tag{6}$$

where β^* is obtained using generalized least-squares estimation

$$\beta^* = \left(\mathbf{1}^T \mathbf{R}^{-1} \mathbf{1}\right)^{-1} \mathbf{1}^T \mathbf{R}^{-1} \mathbf{y}, \tag{7}$$

and $\mathbf{1}$ is a vector filled with ones, and $\mathbf{R}$, $\mathbf{r}(\mathbf{x})$ are the correlation matrix and the correlation vector, respectively. $\mathbf{R}$ and $\mathbf{r}(\mathbf{x})$ are defined as follows:

$$\mathbf{R} = \begin{bmatrix} R\left(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}\right) & R\left(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}\right) & \cdots & R\left(\mathbf{x}^{(1)}, \mathbf{x}^{(m)}\right) \\ R\left(\mathbf{x}^{(2)}, \mathbf{x}^{(1)}\right) & R\left(\mathbf{x}^{(2)}, \mathbf{x}^{(2)}\right) & \cdots & R\left(\mathbf{x}^{(2)}, \mathbf{x}^{(m)}\right) \\ \vdots & \vdots & \ddots & \vdots \\ R\left(\mathbf{x}^{(m)}, \mathbf{x}^{(1)}\right) & R\left(\mathbf{x}^{(m)}, \mathbf{x}^{(2)}\right) & \cdots & R\left(\mathbf{x}^{(m)}, \mathbf{x}^{(m)}\right) \end{bmatrix} \in \mathbb{R}^{m \times m}, \mathbf{r}(\mathbf{x}) = \begin{bmatrix} R\left(\mathbf{x}^{(1)}, \mathbf{x}\right) \\ R\left(\mathbf{x}^{(2)}, \mathbf{x}\right) \\ \vdots \\ R\left(\mathbf{x}^{(m)}, \mathbf{x}\right) \end{bmatrix} \in \mathbb{R}^m, \tag{8}$$

where $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(j)}$ are the i−th and j−th sample points, respectively. $\mathbf{x}$ is an untried point that we want to predict.

Moreover, the kriging model can provide an estimate of the variance of the prediction. It is of the form

$$s^2(\mathbf{x}) = \hat{\sigma}^2\left(1-\mathbf{r}(\mathbf{x})^T\mathbf{R}^{-1}\mathbf{r}(\mathbf{x})\right) + \hat{\sigma}^2\frac{\left(1-\mathbf{r}(\mathbf{x})^T\mathbf{R}^{-1}\mathbf{r}(\mathbf{x})\right)^2}{\mathbf{1}^T\mathbf{R}^{-1}\mathbf{1}} \quad (9)$$

where the parameter $\hat{\sigma}^2$ is

$$\hat{\sigma}^2 = \frac{1}{m}\left(\mathbf{y}-\mathbf{1}\beta^*\right)^T\mathbf{R}^{-1}\left(\mathbf{y}-\mathbf{1}\beta^*\right) \quad (10)$$

2.2 Main steps for building a kriging model

Although the basic theory of kriging has been expressed above, the process for building a kriging model still seems unclear. For a better understanding, the main steps for building a kriging model are summarized as follows:

Step 1: Provide the sample points $\mathbf{X}$ and the corresponding response values $\mathbf{y}$. Determine which spatial correlation function to use (e.g., Gaussian exponential correlation function).

Step 2: Estimate the unknown hyper-parameters in spatial correlation function by maximizing the concentrated likelihood function.

Step 3: Calculate the prediction (6) and the associated estimation error (9) at untried points.

For a specific problem with given training data, the predictive performance of the kriging model is associated with the values of hyper-parameters (Hollingsworth and Mavris 2003). There is no analytical solution for the hyper-parameters and an optimizer is often used to estimate the hyper-parameters. A better set of optimum values for hyper-parameters allow the kriging model to better represent the true response of the objective function. Figure 1 shows an example of (a) an optimized hyper-parameter and (b) an overestimated hyper-parameter. From this example, we can find that estimating the hyper-parameters is very important when constructing the kriging model.

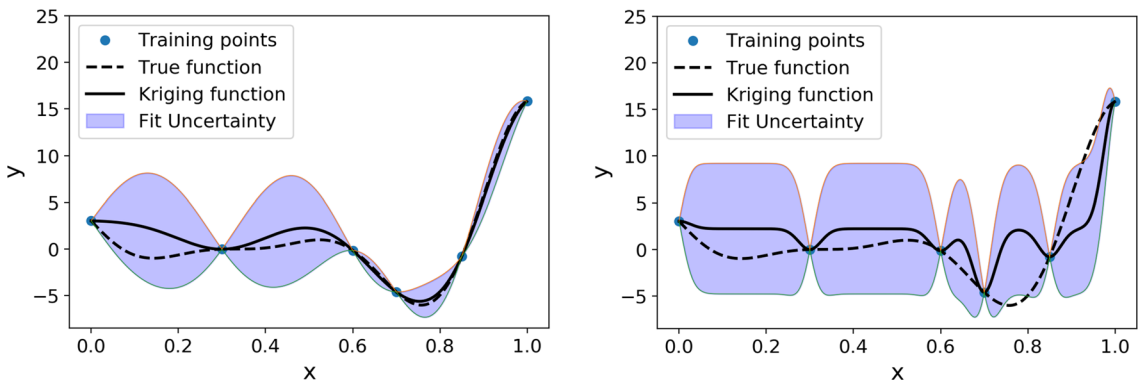
In practice, the hyper-parameters are often estimated by maximizing concentrated likelihood function. The optimization sub-problem can be formulated as follows:

$$\theta = \arg\ \max\left(-\frac{m}{2}\ln\hat{\sigma}^2-\frac{1}{2}\ln|\mathbf{R}|\right), \quad (11)$$

where $|\mathbf{R}|$ denotes the determinant of the correlation matrix. The concentrated likelihood function only depends on the hyper-parameters. For more details of the derivation of the concentrated likelihood function, see, for instance, Jones (2001). The likelihood function is often multimodal and the gradient is difficult to calculate. To avoid becoming trapped in a local maximum, evolutionary algorithms are often used to solve this optimization sub-problem, such as genetic algorithm (Forrester et al. 2008) and differential evolution (Chen et al. 2019). The evolutionary algorithms typically require tens of thousands fitness evaluations of the concentrated likelihood function. However, as mentioned in Section 1, each evaluation of the likelihood function will be computationally expensive for high-dimensional problems. Therefore, the Step 2 is the longest process when building a kriging model. If we can reduce the number of parameters, we need to optimize in this process, the modeling efficiency of kriging will be significantly improved. This proposition has been validated by Bouhlel et al. (2016).

2.3 Relationship between optimized hyper-parameters and global sensitivities

Hyper-parameters are core parameters for kriging model. From the previous studies, we can find that the optimized hyper-parameter θ_l in kriging is a good indicator of the global



(a) Optimized hyper-parameter (b) Overestimated hyper-parameter

Fig. 1 Example of a 1-D function to show the importance of hyper-parameter optimization. The blue space is the uncertainty of the kriging model. **a** Optimized hyper-parameter. **b** Overestimated hyper-parameter

sensitivity of lth input variable (Forrester et al. 2008; Forrester and Keane 2009; Chen et al. 2019). The novel kriging modeling method proposed in this study is based on this conclusion. Therefore, it is essential to provide a brief introduction about this conclusion before discussing our approach. First of all, the global sensitivity analysis technique and Sobol' indices are briefly revisited in this section.

Sensitivity analysis is a technique that studies how the variability of a function's output responds to changes of its inputs variables (Shan and Wang 2010). Sensitivity analysis includes local and global sensitivity analysis. The local sensitivity analysis often focuses the local variability of the output at a given point, which is usually based on the derivative and can be easily calculated (Haftka and Mroz 1986). While global sensitivity analysis allows input variables varying in their whole distribution ranges, which provides an overall view of the impact of input variables on the output (Saltelli et al. 2008). In the last several decades, many global sensitivity analysis methods have been proposed. Sobol' indices (Sobol 2001) are one of the classical global sensitivity analysis methods, which based on variance decomposition. If the number of sample points is enough, Sobol' indices can identify the accurate influence of input on output for any type of functions. Therefore, Sobol' indices are used to calculate the global sensitivities in this research. According to the theory of analysis of variance, an integrable function $f(\mathbf{x})$ defined in $\mathbf{I}^d$ can be decomposed as follows:

$$f(\mathbf{x}) = f_0 + \sum_{i=1}^{d} f_i(x_i) + \sum_{i<j}^{d} f_{ij}(x_i, x_j) + \cdots + f_{12\cdots d}(x_1, x_2, \cdots, x_d). \tag{12}$$

Then the total variance of $f(\mathbf{x})$ can be defined as follows:

$$D = \int_{I^d} f^2(\mathbf{x}) d\mathbf{x} - f_0^2, \tag{13}$$

which can be calculated as sum of partial variances as follows

$$D = \sum_{i}^{d} D_i + \sum_{i<j}^{d} D_{ij} + \cdots D_{12\cdots d}, \tag{14}$$

where the partial variances are computed from each of the terms in Eq. (12) as follows:

$$D_{ij\cdots h} = \int f_{ij\cdots h}^2 dx_i dx_j \cdots dx_h \text{ for } 1 \le i < j < \cdots < h \le d. \tag{15}$$

Using Eq. (14) and Eq. (15), the first-order Sobol' index S_i is defined as follows:

$$S_i = \frac{D_i}{D} \text{ for } 1 \le i \le d. \tag{16}$$

Figure 2 a shows a specific example to visualize the relationship between optimized hyper-parameter θ_l and global sensitivity index S_l. For this function, the functional change almost depends on x_2 and x_1 has little effect (see Fig. 2a). The optimized hyper-parameter θ_2 is much larger than θ_1 and the Sobol indices show that variable x_2 has a higher global sensitivity for the output. It seems that there is a monotonic relationship between the optimized hyper-parameter θ_l and the Sobol' index S_l. To validate this proposition, a 20-D Ellipsoid function is used here.

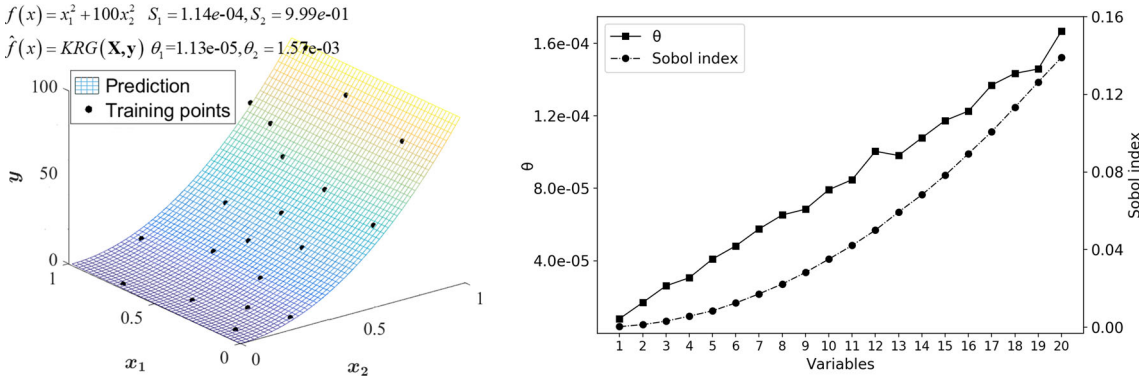
$$f(x) = \sum_{i=1}^{20} i x_i^2, x_i \in [-5, 5], i = 1, ..., 20. \tag{17}$$

This function is suitable for demonstration, because the global sensitivity of each variable on the output is intuitionistic (for $1 \le i < j \le 20$,, the global sensitivity of x_j is higher than that of x_i). Thus, the relationship between optimized hyper-parameter θ_l and global sensitivity index S_l can be clearly observed (see Fig. 2b). As a consequence, we can conclude that the optimized hyper-parameter θ_l in kriging is a good indicator of the global sensitivity of lth input variable. Actually, this conclusion has been demonstrated by Forrester et al. (2008).

Here, an explanation about how the choice of hyper-parameter θ_l affects the output of the kriging model is provided. In kriging, the spatial correlation function is used for calculating the correlation between two sample points. Taking the Gaussian exponential correlation function (Eq. 5) as an example, there are d hyper-parameters $(\theta_1, \theta_2, \cdots, \theta_d)$ corresponding to the d design variables. Each hyper-parameter can affect how far a sample point's influence extends. As shown in Fig. 3, a small θ_l means that all points will have a high correlation and the function values are similar across the sample points in the lth coordinate direction. The higher values of θ_l denote that function values can change rapidly over a small distance in the lth coordinate direction. In other words, a larger value of θ_l means that the lth variable has a higher global sensitivity. This feather of hyper-parameters has been discussed by many researchers (Jones 2001; Forrester et al. 2008; Ulaganathan et al. 2016b).

3 Kriging model combined with MIC

The motivation for this research is to develop a novel kriging modeling method which can build a high-dimensional kriging model with a little amount of computational effort. One of the main challenges for high-dimensional kriging model is due to the huge searching space when optimizing hyper-parameters. As explained above, there is a monotonic relationship between Sobol' indices and optimized hyper-parameter. If we can obtain the global sensitivities of each input variable, we will know the relative magnitude of the optimized hyper-



(a) A visual example (b) A 20-dimensional Ellipsoid function

Fig. 2 Examples to show the relationship between optimized hyper-parameters and global sensitivities. **a** A visual example. **b** A 20-dimensional Ellipsoid function

parameters before optimizing hyper-parameters. Using this knowledge reasonably, we could search the hyper-parameters at a small region instead of the original high-dimensional space. Then the modeling efficiency can be effectively improved. However, obtaining Sobol' indices requires Monte Carlo simulation and is very computationally expensive (Lee et al. 2019). In the last several decades, many global sensitivity analysis methods have been proposed. After an extensive investigation, we find that dependence measure can also be used for global sensitivity analysis (Da Veiga 2015) and MIC (Reshef et al. 2011) is a powerful dependence measure method which is touted as a "correlation for the 21st century" (Speed 2011). Many factors make MIC suitable for the purpose of estimating the relative magnitude of optimized hyper-parameters: (1) MIC can measure the dependence between variables and can be utilized for global sensitivity analysis. According to the proposition discussed in Section 2.3, there is a relationship between MIC values and optimized hyper-parameters; (2) MIC does not relay on the distributional assumptions of data; (3) MIC could identify both linear and non-linear dependencies between variables; (4) Better yet, the MIC values are easy to compute.

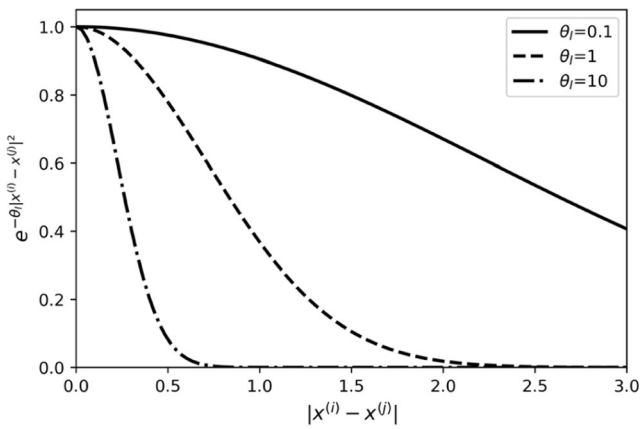


Fig. 3 Relationship between correlation and hyper-parameter θ_l

In this section, a novel kriging modeling method KMIC is developed to improve the modeling efficiency of kriging in high-dimensional problems. The main steps of KMIC are listed as follows:

1. Use MIC to compute the global sensitivities of each input variable.
2. Assume MIC values can be used to estimate the relative magnitude of optimized hyper-parameters.
3. Define a new maximum likelihood estimation problem by using MIC values.
4. Optimize the unknown parameters in the new optimization sub-problem.

The key steps of KMIC are introduced in the following. Besides, a numerical function is chosen to validate the performance of KMIC.

3.1 Maximal information coefficient for global sensitivity analysis

MIC is an optimized version of mutual information (MI). Before we introduce the MIC, we need to review the MI first. MI, based on concepts from information theory, is a measure of how much information two variables share. The value of MI ranges from 0 to +∞. A larger value of MI means a larger amount of information about one random variable obtained through the other random variable. The definition of MI between two random variables is given by

$$I(X;Y)=\int_x\int_y p(x,y)\log\frac{p(x,y)}{p(x)p(y)}dxdy, \tag{18}$$

where $p(x)$ and $p(y)$ are the probability density functions of X and Y, and $p(x,y)$ is the joint probability density function.

In this article, the design variables and output variable are continuous, while we only have finite sample data. Hence, the

density functions of these variables are all unknown. A simple approach to estimate the probability density function is to discrete the data (Kinney and Atwal 2014). By superimposing a rectangular grid on the scatterplot of two variables x and y, each continuous x value (or y value) to the column bin (or row bin) can be assigned into which it falls. Then the estimated density functions $\hat{p}$ can be computed by simply counting the data points falling into each bin. Taking the lth variable and the output for example, the MI value of these two variables is often estimated by the following formula in practice.

$$\hat{I}(\mathbf{x}_l;\mathbf{y}) = \sum_{i=1}^{m} \hat{p}\left(\mathbf{x}_l^{(i)},\mathbf{y}^{(i)}\right)\log\frac{\hat{p}\left(\mathbf{x}_l^{(i)},\mathbf{y}^{(i)}\right)}{\hat{p}\left(\mathbf{x}_l^{(i)}\right)\hat{p}\left(\mathbf{y}^{(i)}\right)}. \tag{19}$$

The basic idea of MIC is that the scatterplot of data $\mathbf{x}_l$ and $\mathbf{y}$ can be gridded by a row and b column and the data $x_l^{(i)}$ ($y^{(i)}$) can be assigned to the row (column) bin it belongs to. Then the MI value for this grid is estimated. To calculate the MIC value of two variables, various possible grids are explored. Then the possible MI values are achieved and these MI values are normalized to make the values from different grids comparable. Different from MI, the MIC ranges from 0 to 1. The MIC of two variables $\mathbf{x}_l$ and $\mathbf{y}$ is defined as follows:

$$MIC(\mathbf{x}_l;\mathbf{y}) = \max_{a,b<B}\frac{\hat{I}(\mathbf{x}_l;\mathbf{y})}{\log_2(\min(a,b))}, \tag{20}$$

where B is the upper bound of the grid size and is a function of sample size m. The authors of MIC suggest $B = m^{0.6}$ (Reshef et al. 2011).

After MIC and its algorithm were published, several researchers have utilized the MIC for feature selection or feature screening (Zhao et al. 2013; Sun et al. 2018; Hemmateenejad and Baumann 2018). Feature selection or feature screening is an important topic of machine learning research (Chen et al. 2019). These applications demonstrate that MIC is an efficient method for identifying the variables with significant influence on the output. Thus, MIC is used in this study to compute the global sensitivities of each design variable. The Python package minepy[1] provides an implementation of MIC (Albanese et al. 2013). In this study, we use this package with default parameters to calculate the MIC values.

3.2 An example to show the relationship between optimized hyper-parameters and MIC values

According to the description above, given a set of sample data about a nonlinear function $y = f(\mathbf{x})$, both the optimized hyper-parameter θ_l in kriging and the MIC value can be used as an indicator of the global sensitivity of lth input variable. It is reasonable to believe that there exists a relationship between the optimized hyper-parameters values and MIC values. To have a better understanding of this relationship, the g07 function (Michalewicz and Schoenauer 2014) with 10 dimensions is used here. The experiment is conducted at a PC with Intel Core i7-2600 CPU @ 3.40GHz and 8 GB RAM.

$$\begin{aligned} f(\mathbf{x}) = {} & x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3-10)^2 + 4(x_4-5)^2 + (x_5-3)^2 \\ & + 2(x_6-1)^2 + 5x_7^2 + 7(x_8-11)^2 + 2(x_9-10)^2 + (x_{10}-7)^2 + 45, \\ & x_i \in [-10, 10], i = 1, \ldots, 10. \end{aligned} \tag{21}$$

First of all, we generate a 100×10 matrix of observed points $\mathbf{X}$ by Latin hypercube sampling (Mckay et al. 1979). Then obtain the 100×1 responses vector $\mathbf{y}$. Before calculating the hyper-parameters and MIC values, data pre-treatment is carried out, where the data $\mathbf{X}$ and $\mathbf{y}$ are centered to have zero mean.

Secondly, calculate the MIC values of each input variable and output variable. The MIC values for this design of experiment are 0.22, 0.24, 0.23, 0.33, 0.22, 0.21, 0.23, 0.81, 0.25, and 0.22 from $(\mathbf{x}_1, \mathbf{y})$ to $(\mathbf{x}_{10}, \mathbf{y})$. It takes 0.024 s to calculate these values.

Next, build the ordinary kriging model with the observed points $\mathbf{X}$ and the corresponding response values $\mathbf{y}$. To estimate the hyper-parameters, an effective differential evolution (DE) algorithm, jDE (Brest et al. 2006), is used. DE is a popular global optimization algorithm and there are quite a few different DE variants. In this article, we use DE/rand-to-best/1 to generate new solutions. The population size is set as 100, and the maximum number of function evaluations is 10,000. Then the unknown hyper-parameters are searched in the range of $[10^{-6}, 10^{2}]$. Despite DE is a stable algorithm with high performance, we cannot guarantee that we can find the global maximum of the likelihood function with given computational effort. Thus, the process of training the kriging model is repeated 20 times. Each experiment takes about 13 s to estimate the hyper-parameters. The best result in these experiments is

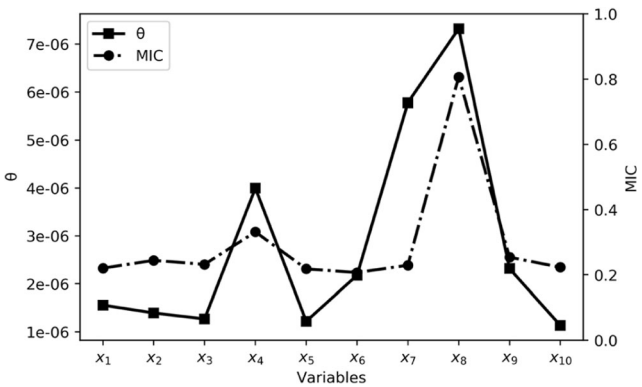


Fig. 4 The hyper-parameters and MIC values for the g07 function

[1] http://minepy.readthedocs.io/

recorded. The recorded hyper-parameters corresponding to the ten variables are 1.56e-06, 1.39e-06, 1.27e-06, 4.00e-06, 1.22e-06, 2.18e-06, 5.78e-06, 7.32e-06, 2.32e-06, and 1.13e-06, respectively.

Finally, we plot the hyper-parameters and MIC values in Fig. 4.

As can be seen in Fig. 4, the trends of the optimized hyper-parameters and MIC values are very similar. From the expression of the g07 function, we can see that variable x_8 has the largest coefficient and should be the most important variable (the variable with the highest global sensitivity). In practice, both hyper-parameters and MIC values show that the influence of variable x_8 on the output is the most significant. The coefficients of five variables x_1, x_2, x_3, x_5 and x_{10} are small. Similarly, the hyper-parameters corresponding to these five variables are smaller than the hyper-parameter corresponding to variable x_8. It is obvious that the magnitude of hyper-parameters can reflect the global sensitivities of each design variable. Unfortunately, MIC cannot clearly distinguish the variables except variable x_4 and variable x_8 in this case. Hyper-parameter values show variable x_7 is the second most important variable, while the MIC values show that variable x_7 is less important than variable x_4. However, it should be noted that the time for estimating hyper-parameters is about 542 times longer than that for calculating the corresponding MIC values. MIC cannot capture variables x_6, x_7 and x_9, but the MIC values for these variables are not much lower than the MIC value of variable x_1 and output. Therefore, MIC is an alternative method for quickly estimating the relative magnitude of optimized hyper-parameters with less loss of information.

3.3 Construction of new maximum likelihood estimation problem for KMIC model

As explained above, if the number of design variables is large, maximum likelihood estimation problem itself is a high-dimensional optimization sub-problem when constructing kriging model. If we can improve the efficiency for solving this optimization sub-problem, the time for constructing kriging model can be significantly reduced. The question is how to speed up the process of solving this optimization sub-problem. In recent years, knowledge-assisted optimization becomes an interesting research topic for tackling high-dimensionality optimization problems (Wu et al. 2017; Wu and Wang 2018). By incorporating existing knowledge into optimization, the optimization efficiency can be improved. Inspired by this idea, we take the relationship between optimized hyper-parameters and global sensitivities into consideration when maximizing the likelihood function.

Taking the g07 function as an example, we actually have some knowledge about the hyper-parameters before maximizing the likelihood function. Variable x_8 is the most important variable for the g07 function. Thus, the hyper-parameter corresponding to variable x_8 should be the biggest. Variables x_1, x_2, x_3, x_5 and x_{10} are less important than variable x_8 but cannot be excluded. Compared with variable x_8, the hyper-parameters corresponding to these five variables should be smaller but will not be too small. These knowledge can be taken into consideration when estimating the hyper-parameters. However, when constructing kriging model for a specific problem, the design function is often a black-box. We do not know the coefficient of each variable and will not know which variable is more important. Fortunately, MIC can be used to

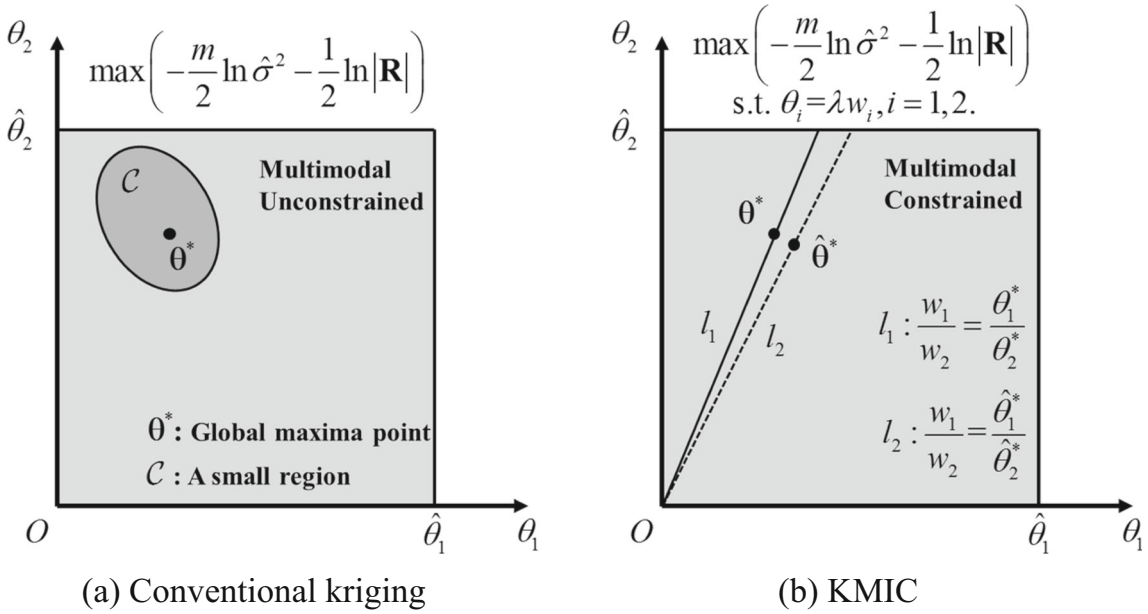


Fig. 5 Maximum likelihood estimation problem. **a** Conventional kriging. **b** KMIC

Table 1 Comparison results for g07 function with 100-point

	Statistic	Times(s)	R^2	NRMSE	NMAE
OK	Mean	12.7947	1.0000	9.3148e−06	1.0630e−04
	Std	0.3640	1.3398e−10	2.3200e−06	3.3265e−05
KMIC	Mean	0.0724	1.0000	1.4148e−05	1.8897e−04
	Std	0.0044	2.5795e−10	4.0723e−06	7.0258e−05

estimate the relative magnitude of optimized hyper-parameters. This has been discussed in Section 3.2. Therefore, for a black-box problem, the MIC values can be regarded as the existing knowledge that can help to improve the optimization efficiency of the maximum likelihood estimation.

How can we incorporate the known knowledge into the optimization? In order to improve the optimization efficiency as much as possible, a simple strategy is to assume the relationship between hyper-parameters and MIC values is a linear proportional relationship. Then we can add a set of equality constraints to the maximum likelihood estimation problem. The new optimization sub-problem can be expressed as follows:

$$\begin{aligned} \boldsymbol{\theta} &= \arg\ \max\left(-\frac{m}{2}\ln\hat{\sigma}^2-\frac{1}{2}\ln|\mathbf{R}|\right), \\ &s.t.\ \ \theta_i = \lambda w_i, i = 1, 2, ..., d \end{aligned} \tag{22}$$

where λ is an auxiliary parameter and w_i is the MIC value of $\mathbf{x}_i$ and $\mathbf{y}$. For any problems, $\theta_i \in \mathbb{R}^+$, $w_i \in [0, 1]$ and $\lambda \in \mathbb{R}^+$. In practice, a set of boundary constraints should be given when using an optimizer to estimate the auxiliary parameter. The authors suggest searching the unknown auxiliary parameter λ in the range of $[10^{-6}, 100]$ for high-dimensional problems.

In this way, our new optimization sub-problem for estimating hyper-parameters has only one unknown parameter. However, there is an obvious drawback to this approach. Since we bind the original variables together using an auxiliary parameter, their values cannot be changed independently of each other anymore. This limits the reachable solutions in the original search space to get a better result.

For a better understanding of our approach, a geometric explanation is provided here. For the conventional kriging, the concentrated likelihood function is a multimodal, unconstrained function. As shown in Fig. 5a, the global maxima point $\boldsymbol{\theta}^*$ locates at comparatively very small regions in the d−dimensional space. However, to find the maximizer, we have to search in the whole d−dimensional space. As the dimensionality increases, the search space will become vaster and vaster and this optimization problem will be more and more complex. Therefore, it is very time-consuming to solve Eq. (11) for the high-dimensional problem.

Different from the conventional kriging, KMIC estimates the hyper-parameters by solving the new maximum likelihood estimation problem. The new optimization sub-problem is a multimodal, constrained optimization problem. As shown in Fig. 5b, we can think that the new optimization sub-problem has still the same number of design variables as the original one. Owing to these equality constraints, KMIC only needs to search along a straight line in the vast space when solving Eq. (22). The slope of this line is obtained by taking the features of hyper-parameters into consideration. Assume the relationship between optimized hyper-parameters and MIC values is a linear proportional relationship. Then KMIC searches along the line l_1 when maximizing the likelihood function, and KMIC can easily find the global maxima point $\boldsymbol{\theta}^*$. Unfortunately, it is difficult to guarantee this assumption. From the example shown in Section 3.2, it is found that there are differences between the hyper-parameters and MIC values. Actually, KMIC searches the maximizer along the line l_2. For most cases, KMIC will obtain an approximate maximizer at $\hat{\boldsymbol{\theta}}^*$. Despite there is loss of accuracy, it should be noted that KMIC is more efficient than conventional kriging for high-dimensional problem. Besides, the likelihood function around the global maxima point is often flat and there is no point to find the maximizer with great accuracy when constructing the kriging model (Lophaven et al. 2002). Empirical studies show that the accuracy loss of KMIC is acceptable for problems with 40 or more design variables. Therefore, KMIC can provide an alternative way for high-dimensional kriging modeling.

Table 2 Numerical test functions

Name	d	m	Expression
Ellipsoid	20	200	$f(x) = \sum_{i=1}^{20} ix_i^2, x_i \in [-5, 5], i = 1, ..., 20$
Dixon-Price	30	300	$f(x) = (x_1-1)^2 + \sum_{i=2}^{30} i\left(2x_i^2-x_{i-1}\right)^2, x_i \in [-10, 10], i = 1, ..., 30$
Rosenbrock	40	400	$f(x) = \sum_{i=1}^{39}\left[100\left(x_{i+1}-x_i^2\right)^2 + (x_i-1)^2\right], x_i \in [-5, 10], i = 1, ..., 40$
Griewank	50 80	400 500	$f(x) = \sum_{i=1}^{d}\frac{x_i^2}{4000} - \prod_{i=1}^{d}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1, x_i \in [-5, 5], i = 1, ..., d$

Table 3 Optimizer settings for maximizing the likelihood function

Model	Optimization algorithms	Initial coefficient	Max. NFE	Interval
OK	jDE		$1000d$	$[10^{-6}, 100]$
KMIC	COBYLA	$\lambda = m^{-1/m}$	1000	$[10^{-6}, 100]$
KPLS1	COBYLA	$\theta = m^{-1/m}$	1000	$[10^{-6}, 100]$
KPLS2	COBYLA	$\theta_i = m^{-1/m}$, $i = 1, 2$	2000	$[10^{-6}, 100]$
KPLS3	COBYLA	$\theta_i = m^{-1/m}$, $i = 1, 2, 3$	3000	$[10^{-6}, 100]$

3.4 Algorithm implementation

We summarize the implementation of the proposed KMIC model in the following:

Step 1: Provide the sample points **X** and the corresponding response values **y**. Determine which spatial correlation function to use (e.g., Gaussian exponential correlation function).
Step 2: Calculate the MIC values of each input variable and output variable.
Step 3: Estimate the unknown auxiliary parameter by using the numerical optimization algorithm to solve the new maximum likelihood estimation problem.
Step 4: Calculate the approximate hyper-parameters.
Step 5: Calculate the prediction (6) and the associated estimation error (9) at untried points.

3.5 An example for KMIC modeling demonstration

To validate the performance of KMIC, we build KMIC model and ordinary kriging model using the same sample points. The experiment is repeated 20 times and each experiment has a different set of sample points obtained by Latin hypercube sampling with "maximin" criterion. To measure the prediction accuracy of these surrogates, 5000 validation points are randomly selected by Latin hypercube sampling, and the coefficient of determination R^2, normalized root-mean-square error (NRMSE), and normalized maximum absolute error (NMAE) are calculated.

$$R^2 = 1 - \sum_{i=1}^{N} \left(y_i - \hat{y}_i\right)^2 \Big/ \sum_{i=1}^{N} \left(y_i - \bar{y}\right)^2, \tag{23}$$

$$\mathrm{NRMSE} = \sqrt{\sum_{i=1}^{N} \left(\frac{y_i - \hat{y}_i}{y_i}\right)^2 \Big/ N} \tag{24}$$

$$\mathrm{NMAE} = \max\left(\left|\frac{y_i - \hat{y}_i}{y_i}\right|\right) \tag{25}$$

where N is the number of validation points, y_i is the true response of the ith validation point, $\hat{y}_i$ is the predicted value of the ith validation point, and $\bar{y}$ is the mean of true response. R^2 and NRMSE can reflect the global accuracy of the surrogates. NMAE is a criterion that can represent the local predicting performance of the surrogates. The closer the value of R^2 is to 1, the better the model fits the data. The smaller values of the NRMSE and the NMAE, the more accurate the model is.

The results are listed in Table 1. The abbreviation "OK" refers to the ordinary kriging. From the results, we can see that the modeling time is drastically reduced when using the KMIC. More precisely, the ordinary kriging requires an average time of 12.7947 s whereas KMIC requires only 0.0724 s.

Table 4 CPU time for numerical test functions

Surrogate	Statistic	Ellipsoid (20-D)	Dixon-Price (30-D)	Rosenbrock (40-D)	Griewank (50-D)	Griewank (80-D)
OK	Mean	*103.92 s*	*418.44 s*	*1229.86 s*	*1875.72 s*	*5670.74 s*
	Std	1.14 s	8.45 s	4.19 s	8.34 s	37.29 s
KMIC	Mean	*0.35 s*	*0.99 s*	*2.30 s*	*2.86 s*	*6.10 s*
	Std	1.61e−02 s	3.36e−02 s	5.78e−02 s	5.77e−02 s	0.11 s
KPLS1	Mean	*5.78e−02 s*	*0.11 s*	*0.24 s*	*0.26 s*	*0.39 s*
	Std	8.93e−03 s	2.02e−02 s	1.41e−02 s	2.43e−02 s	1.78e−02 s
KPLS2	Mean	*0.35 s*	*0.31 s*	*0.49 s*	*0.49 s*	*0.66 s*
	Std	0.96 s	0.23 s	4.95e−02 s	5.32e−02 s	5.40e−02 s
KPLS3	Mean	*2.13 s*	*2.59 s*	*1.39 s*	*0.73 s*	*1.04 s*
	Std	3.28 s	5.21 s	2.24 s	7.63e−02 s	0.22 s

Table 5 Metrics of modeling accuracy for numerical test functions

	R^2	NRMSE	NMAE	R^2	NRMSE	NMAE	R^2	NRMSE	NMAE
	Ellipsoid (20-D)			Dixon-Price (30-D)			Rosenbrock (40-D)		
OK	*0.98*	*3.36e−02*	*0.13*	*0.78*	*0.14*	*0.77*	0.78	0.14	0.79
KMIC	0.83	9.76e−02	0.60	0.64	0.19	1.25	*0.79*	*0.13*	*0.75*
KPLS1	0.26	0.22	1.66	0.21	0.31	2.64	0.63	0.18	1.26
KPLS2	0.43	0.19	1.33	0.34	0.27	2.23	0.70	0.16	1.07
KPLS3	0.54	0.17	1.06	0.38	0.26	2.02	0.70	0.16	1.08
	Griewank (50-D)			Griewank (80-D)					
OK	0.82	5.02e−03	1.58e−02	0.65	8.41e−03	2.88e−02			
KMIC	*0.90*	*3.72e−03*	*1.13e−02*	*0.83*	*5.85e−03*	*1.83e−02*			
KPLS1	0.30	9.97e−03	3.64e−02	0.26	1.23e−02	4.55e−02			
KPLS2	0.46	8.81e−03	3.16e−02	0.39	1.11e−02	4.12e−02			
KPLS3	0.54	8.14e−03	2.83e−02	0.45	1.06e−02	3.74e−02			

A 99.4% saving of time is achieved using our approach. In the respect of model accuracy, R^2 cannot distinguish which model is worse. NRMSE and NMAE show that the accuracy of KMIC is slightly lower. Therefore, with the result of comparison, we can conclude that the efficiency of the ordinary kriging can be improved by KMIC, and the loss of accuracy is acceptable.

4 Experimental study

In this section, to further examine the performance of the proposed method, KMIC is compared with the ordinary kriging and KPLS. KPLS is a recently proposed method which can accelerate the computational process of building the kriging model in high-dimensional problems. By combining the partial least squares (PLS) technique with kriging, the relationship between input variables and output variable is considered and some principal components of the original data are reserved. The KPLS has shown an outstanding result in terms of saving computation time, which is similar with KMIC. It is worth mentioning that the accuracy and modeling time of KPLS are related to the number of principal components. Therefore, the KPLS models with one to three principal components are all considered in this section and these models are denoted by KPLS1, KPLS2, and KPLS3, respectively. Review of the theory and implementation of KPLS are beyond the scope of this article. Interested readers can refer to the literature (Bouhlel et al. 2016; Hartwig and Bestle 2017) for more details. The modeling of KPLS is implemented using the surrogate modeling toolbox (SMT).[2] The experiments for each test function in this section are done in the same experimental environment described in Section 3.2.

4.1 Numerical examples

To demonstrate the efficiency of the KMIC for tacking different kinds of complex, high-dimensional problems, five representative numerical test functions varied from 20-D to 80-D are used here. Descriptions of these test functions are summarized in Table 2.

Similar to the example in Section 3.5, both the sample points and validation points are selected by Latin hypercube sampling. Once the kriging models are constructed, 5000 validation points are randomly selected to assess the model accuracy under three metrics R^2, NRMSE, and NMAE. To analyze the robustness of these methods, the experiments are repeated 20 times. Then the mean and standard deviation values of modeling time and three accuracy metrics are calculate.

In practice, the accuracy of kriging model can be greatly affected by the choice of optimization algorithm for the process of estimating the hyper-parameters. The main characteristics of the optimization algorithms we adopted are shown in Table 3. Similar to the example in Section 3, jDE is used for the ordinary kriging considering its high performance when handling complex, high-dimensional problems. The population size is set as 100. The maximum number of function evaluations (Max. NFE) is $1000d$. The unknown hyper-parameters of the ordinary kriging are searched in the range of $[10^{-6}, 100]$. The KPLS available in SMT optimizes the hyper-parameters by a derivative-free optimization algorithm COBYLA (Powell 1994). Compared with evolutionary algorithms, COBYLA is more suitable to optimize the likelihood function for problems with a small number of design variables. KMIC only needs to optimize one parameter. To be fair, optimizer settings for KMIC are the same as that of the KPLS with one principal component.

[2] https://github.com/SMTorg/smt.

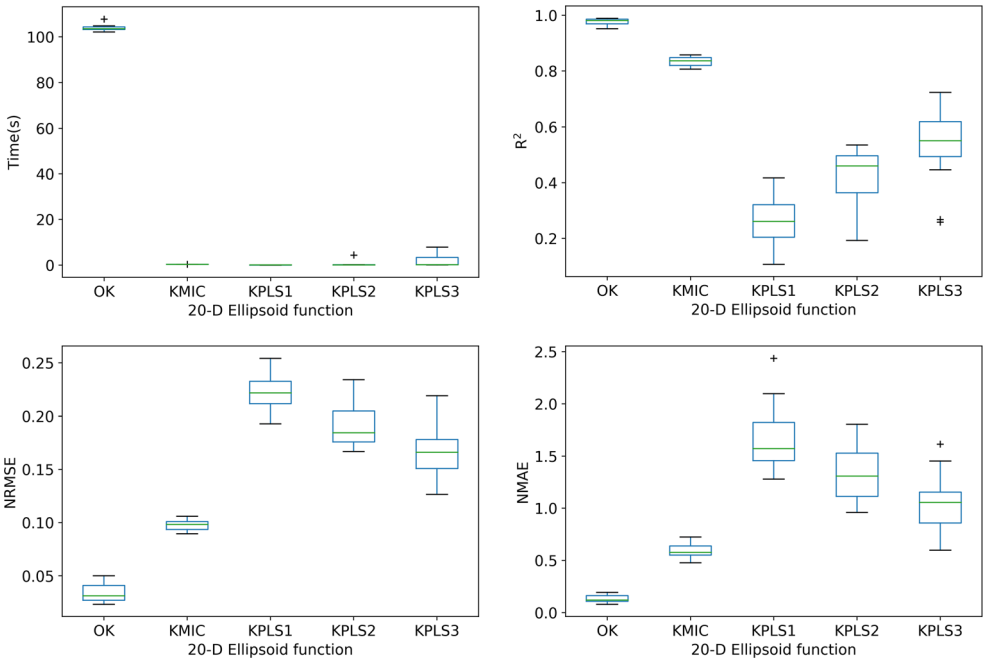


Fig. 6 Box-plots for the 20-D Ellipsoid function

4.2 Results discussion

4.2.1 Comparison of computational efficiency

When it comes to the cases that the kriging model needs to be frequently constructed, the modeling efficiency of kriging model should be a considerable issue. The ideal case is that both the modeling accuracy and the modeling efficiency are as high as possible. But it is difficult to balance the modeling accuracy and modeling efficiency. KMIC is proposed as an attempt to approach this goal. To verify the modeling efficiency of the proposed method, the CPU time for constructing kriging models using different methods was measured as shown in Table 4. The mean values are shown in italics for ease of comparison. From the results, we can find that the modeling time of KMIC is significantly shorter than that of the ordinary kriging in all the test functions. KMIC has achieved a 99.7~99.9% saving of time for constructing the kriging model. The computational cost of KMIC can be negligible for most of the problems in engineering. In addition, both KMIC and KPLS can build a kriging model in several seconds for high-dimensional problems.

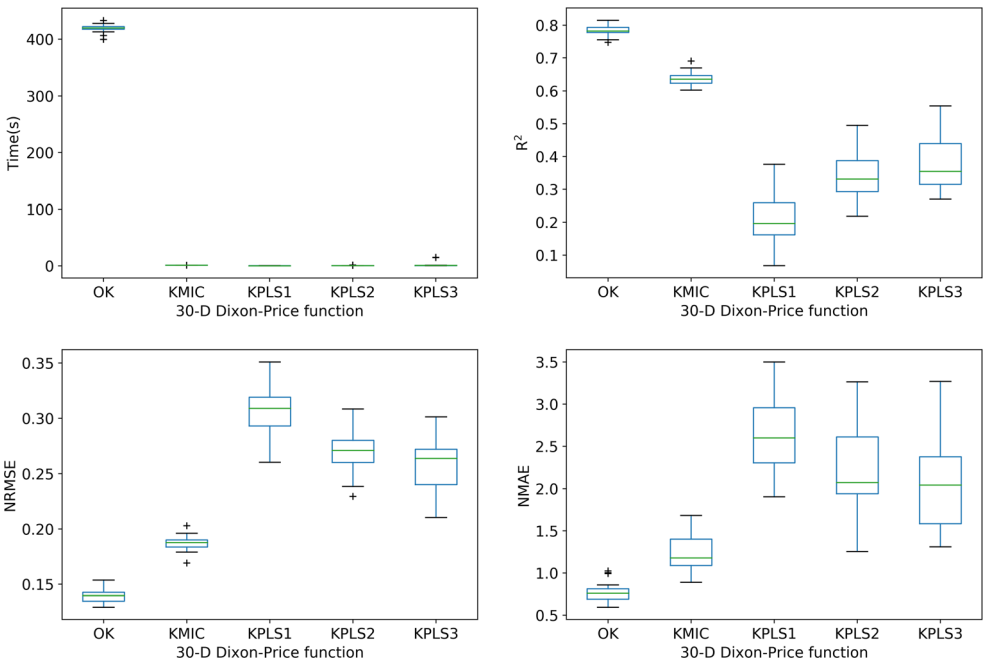


Fig. 7 Box-plots for the 30-D Dixon-Price function

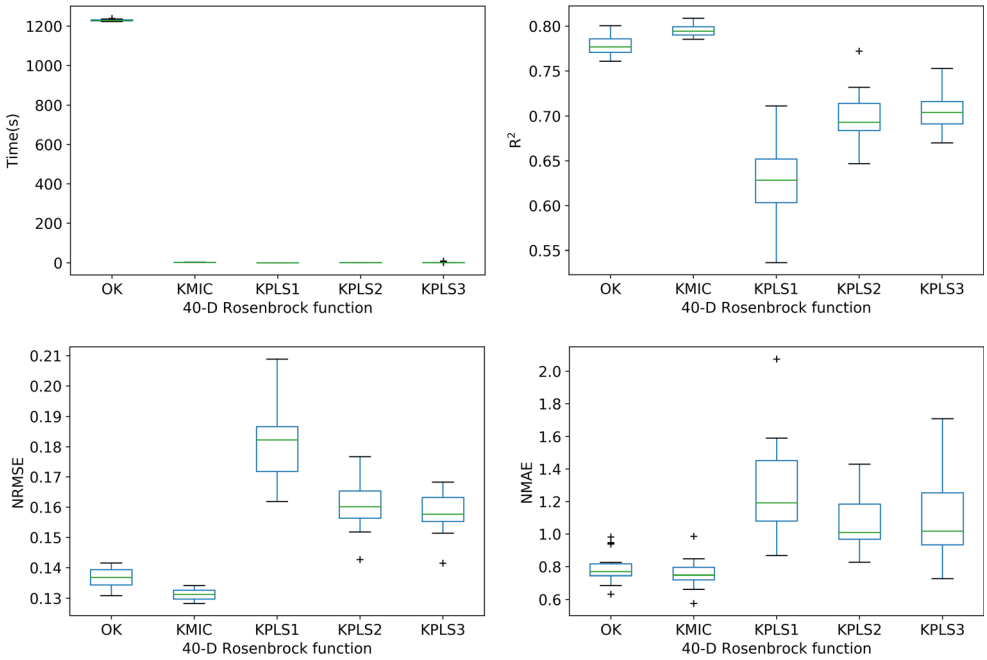


Fig. 8 Box-plots for the 40-D Rosenbrock function

4.2.2 Accuracy comparison

To demonstrate comparative performance of KMIC in terms of accuracy, three metrics R^2, NRMSE, and NMAE are used to assess the global and the local accuracy of the kriging models. The results of the three kriging modeling methods (ordinary kriging, KPLS, and KMIC) on the five test functions are shown in Table 5. For ease of comparison, the mean values of the best results are shown in italics. To better visualize the results, boxplots are used in Figs. 6, 7, 8, 9, and 10. With a glance over the results, the ordinary kriging behaves best for the Ellipsoid function and Dixon-Price function. For problems with 40 or more variables, results show that KMIC is even more accurate than the ordinary kriging. Besides, KMIC outperforms KPLS with one to three principal components for all the test functions.

For the Ellipsoid function, as shown in Table 5 and Fig. 6, three metrics all show that the accuracy of the ordinary kriging performs the best and KPLS is the worst. A larger R^2 means a better fitting of the kriging model. The R^2 of the ordinary kriging and KMIC are 0.98 and 0.83, respectively. This means the global accuracy of KMIC is slightly worse than that of the ordinary

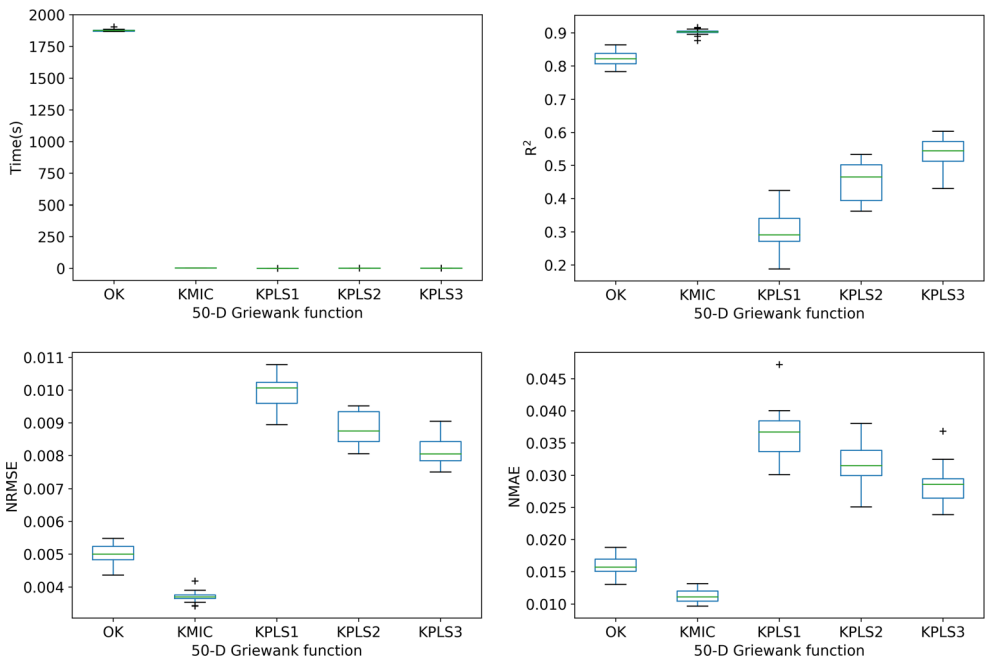


Fig. 9 Box-plots for the 50-D Griewank function

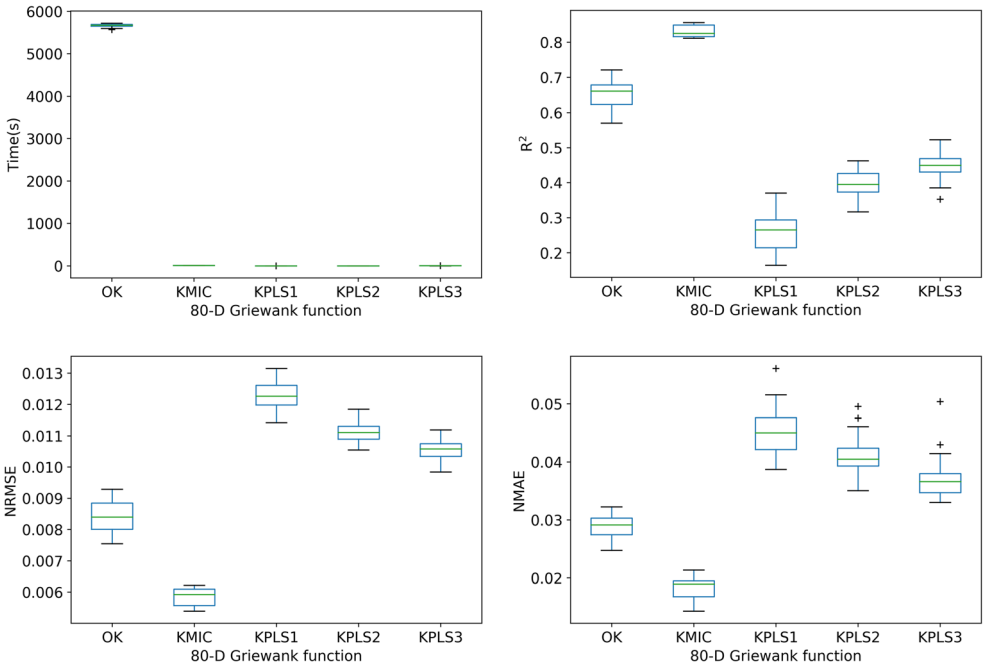


Fig. 10 Box-plots for the 80-D Griewank function

kriging. It can be seen that NRMSE could also reflects the global accuracy of the kriging models. NMAE shows that the ordinary kriging performs best in terms of the local accuracy. However, it should be noted that the modeling time of the ordinary kriging is about 297 times that of KMIC. KMIC can obtain a balance between modeling accuracy and modeling efficiency. The results of the Dixon-Price function and Ellipsoid function are similar (see Table 5 and Fig. 7). The Rosenbrock function is a multimodal problem with a steep, curved valley. As can be seen in Table 5 and Fig. 8, the R^2 of the ordinary kriging and KMIC are 0.78 and 0.79, respectively. The NRMSE and NMAE also show that KMIC is slightly more accurate than the ordinary kriging. For the Griewank functions with 50 and 80 variables, the accuracy of KMIC is the best. For the Griewank function with 80 variables, the global accuracy of KMIC is better than that of the ordinary kriging, with an R^2 of 0.83 for the former and 0.65 for the latter. The NMAE of KMIC is 1.83e-02 which is lower than that of the ordinary kriging. Therefore, the local accuracy of KMIC is better than that of the ordinary kriging for the 80-D Griewank function.

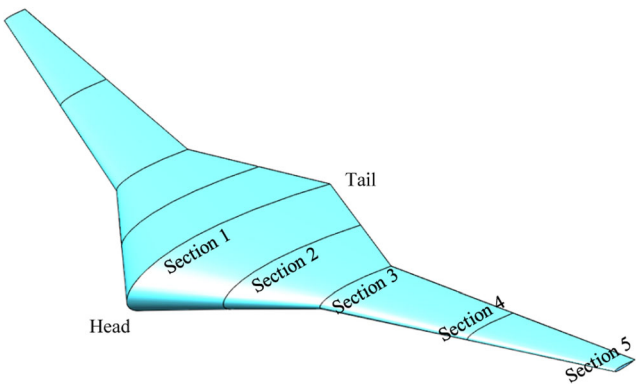


Fig. 11 3-D shape of the blended-wing-body underwater glider

From the above observations, we can find that the proposed method is suitable for tackling high-dimensional problems when the kriging model needs to be frequently constructed. It should be noted that the model accuracy of KMIC is expected to be somewhat worse than that of ordinary kriging. However, for the problems with more than 40 variables, KMIC can obtain a more accurate kriging model than ordinary kriging with given computational effort. The most probable reason is that the process of optimizing hyper-parameters becomes harder as the dimensionality of the test function increases. Under limited computational effort, it is difficult to build an ordinary kriging model with great accuracy. If we do not consider the limit of the maximum number of function evaluations when maximizing the likelihood function, we believe that the ordinary kriging will be more accurate than KMIC for all the test functions. This phenomenon shows that for problems with more than 40 variables, KMIC is better than ordinary kriging in practice. Liu et al. (2014) draw a similar conclusion that the accuracy of direct kriging modeling for problems with 30 or less variables is better.

Another interesting phenomenon is that as the number of training points increases, the time for constructing KMIC becomes longer. This phenomenon is also found for KPLS. Actually, the main reason has been mentioned in Section 1. Despite both KMIC and KPLS1 only need to optimize one parameter when training the kriging model, they require inverting the covariance matrix for several times. If the number of training points increases, inverting the covariance matrix will require more time. The number of training points for Rosenbrock function and 50-D Griewank function is 400,

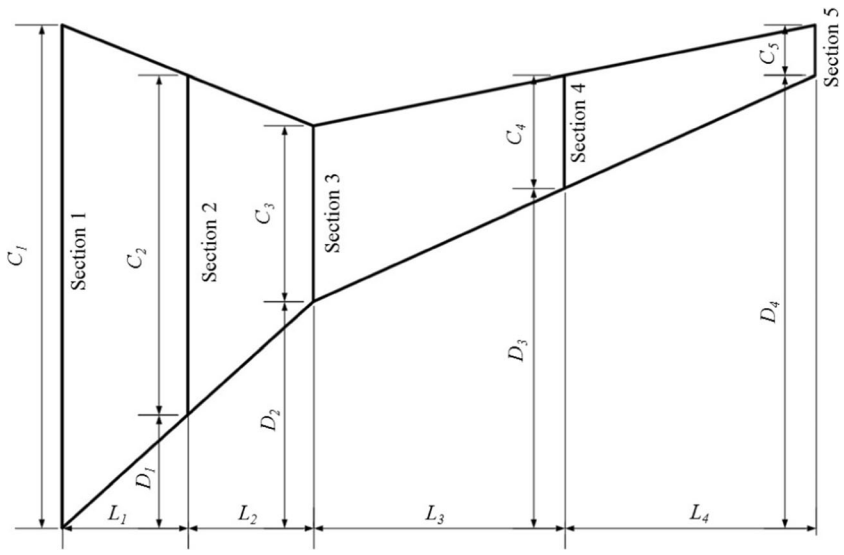


Fig. 12 Geometric parameters of the planar shape

while the 50-D Griewank function requires slightly more time. This is because that as the dimensionality increases, calculating MIC values and utilizing PLS require more computational effort. Besides, calculating MIC values requires more time than utilizing PLS. Thus, KMIC requires more time than KPLS1 in all the test cases.

4.3 Industrial example

In this part, the performance of KMIC is validated through an engineering application. Here, we focus on a blended-wing-body underwater glider and use KMIC to generate kriging model for the drag coefficients (C_d) as function of 35 variables. For the sake of simplicity, a blended-wing-body underwater glider formed with five control sections is used here (see Fig. 11). The planar shape of this underwater glider is determined and is shown in Fig. 12. Geometric parameters of the planar shape are shown in Table 6. The shape of the underwater glider is mainly determined by the five control sections. The five control sections are all symmetrical airfoils. Initially, the standard NACA0022, NACA0019, NACA0016, NACA0014, and NACA0012 airfoils are selected as the section 1, 2, 3, 4, and 5, respectively. The class-shape function transformation (CST) method (Kulfan and Bussoletti 2006) is then used to parameterize each of the control sections. A sixth-order Bernstein polynomial is chosen to fit each sectional airfoil. Each sectional airfoil is represented with 7 design variables. Therefore, there are totally 35 variables describing the shape of the underwater glider. In this research, each of the control sections is confined within another two airfoils. For example, section 1 is confined between the standard NACA0016 and NACA0028 airfoils.

Then a CFD-based numerical simulation process is built to calculate the drag coefficients. The fluid material is water-liquid which is regarded as incompressible fluid with a density of 998.2 kg/m^3, and a dynamic viscosity of 1.003×10^{-3} Pa/s. The k-ω shear stress transport (SST) turbulence model is adopted to solve the 3-D Reynolds-averaged Navier-Stokes control equations. The magnitude of the inlet velocity and the angle of attack are set to 1 m/s and 6°, respectively. Besides, the pressure at the pressure outlet is set to be 0 Pa. The convergence criterion is that the root-mean-square residual is less than 10^{-5} for each equation or the total number of iterations reaches 500. The time for each simulation is about 20 min. Interested readers can refer to the literature (Li et al. 2018) for more details of the CFD-based numerical simulation process.

Table 6 Values of the planform parameters

Notation	Values	Description	Notation	Values	Description
L_1	250 mm	Distance between section 1 and 2	C_4	225 mm	Chord length of section 4
L_2	250 mm	Distance between section 2 and 3	C_5	100 mm	Chord length of section 5
L_3	500 mm	Distance between section 3 and 4	D_1	225 mm	Offset of section 2
L_4	500 mm	Distance between section 4 and 5	D_2	450 mm	Offset of section 3
C_1	1000 mm	Chord length of section 1	D_3	675 mm	Offset of section 4
C_2	675 mm	Chord length of section 2	D_4	900 mm	Offset of section 5
C_3	350 mm	Chord length of section 3			

Table 7 Results for the drag coefficient problem of the underwater glider

	Statistic	Times(s)	R^2	NRMSE	NMAE
OK	Mean	*591.35*	*0.99*	*1.10e−03*	*3.47e−03*
	Std	9.35	1.64e−03	1.34e−04	5.78e−04
KMIC	Mean	*1.45*	*0.99*	*1.54e−03*	*4.50e−03*
	Std	3.49e−02	6.45e−04	3.82e−05	5.10e−04
KPLS1	Mean	*0.13*	*0.98*	*1.73e−03*	*4.84e−03*
	Std	1.12e−02	3.84e−03	2.01e−04	9.62e−04
KPLS2	Mean	*0.29*	*0.99*	*1.49e−03*	*4.20e−03*
	Std	1.56e−02	2.23e−03	1.30e−04	9.23e−04
KPLS3	Mean	*1.16*	*0.99*	*1.40e−03*	*4.34e−03*
	Std	1.03	9.65e−04	6.23e−05	1.09e−03

The number of sample points is set to 350, which is ten times the dimensionality. With the given number of training points, five different training sets are obtained by Latin hypercube sampling and used to construct five surrogates by each of the five methods which are used in Section 4.2. The prediction accuracy of the kriging models is evaluated with R^2, NRMSE, and NMAE at 500 testing samples. The average and standard deviation of the three metrics over the five surrogates are calculated to evaluate the statistical performance. The optimizer settings for estimating the hyper-parameters are the same as the parameters listed in Table 3.

The experimental results are shown in Table 7 and Fig. 13. The mean values in Table 7 are shown in italics for ease of comparison. It can be seen that the accuracy of these five surrogates is comparable. However, the ordinary kriging requires an average time of 591.35 s to training the model. In contrast, the modeling time of KMIC and KPLS can be ignored. In terms of modeling accuracy, ordinary kriging is the best and KPLS1 is the worst. The accuracy of KMIC is slightly worse than that of KPLS2 and KPLS3. It should be note that the accuracy of KPLS is related to the number of principal components. However, there is no rule for determining how much number of principal components should be chose, if we want to achieve a balance between model accuracy and modeling efficiency. In contrast, KMIC is more simplicity.

5 Conclusion

In this article, a novel kriging modeling method (KMIC) is developed for high-dimensional design problems. Before training the kriging model, we use the MIC to estimate relative magnitude of hyper-parameters. Then we reformulate the maximum likelihood estimation problem to improve the modeling efficiency. Based on this, KMIC only needs to optimize one auxiliary parameter when estimating the hyper-parameters. For ease of understanding, a geometric explanation about the reformulated maximum likelihood estimation problem is provided and the g07 function with 10 variables is used to demonstrate the efficiency of KMIC. Five representative numerical test functions varied from 20-D to 80-D and an industrial example with 35 variables is used to study the performance of KMIC. Some conclusions can be drawn as follows.

1. Compared with the conventional kriging, a 99% saving of time can be achieved using our approach. Therefore, KMIC is an efficient kriging modeling method for high-

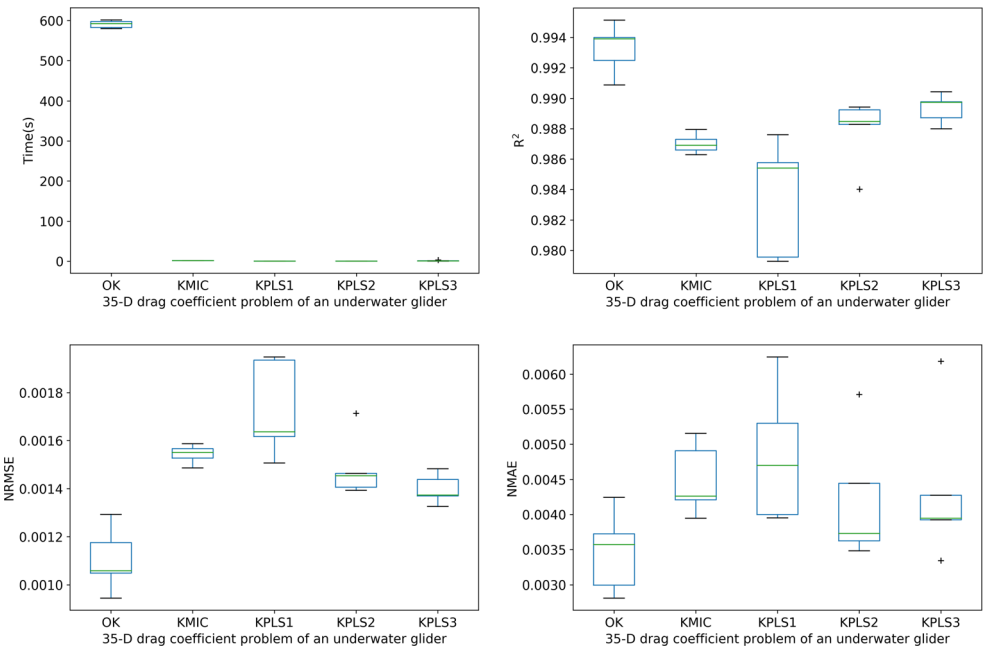


Fig. 13 Box-plots for the 35-D drag coefficient problem of an underwater glider

dimensional problems when the kriging model needs to be frequently constructed.
2. For the problems with 40 or more variables, KMIC is even more accurate than the ordinary kriging with given computational resources. Thus, we recommend KMIC for tackling the design problems with more than 40 design variables.
3. Compared with KPLS, KMIC is more competitive. Thus, KMIC provides a new alternative way for improving the modeling efficiency of kriging.

In our continuous research, the influences of sampling strategies and sample size on the performance of KMIC will be studied. Other verification functions and other types of spatial correlation function will be used to further study the performance of KMIC. Besides, developing some strategies to improve the accuracy of KMIC is an interesting direction for future work.

6 Replication of results

The main steps for constructing the proposed KMIC model are presented in the Section 3.4 to help readers understand better.

Funding information This research was financially supported by the National Natural Science Foundation of China (Grant No. 51875466 and Grant No. 51805436).

Compliance with ethical standards

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

Appendix: Examples of spatial correlation functions

The proposed method (KMIC) combines kriging with maximal information coefficient and constructs a new maximum likelihood estimation problem (Eq. 22). Then the number of parameters we need to optimize when estimating hyper-parameters is reduced to one. It seems that KMIC constructs a new spatial correlation function which depends on only one parameter. In this article, Gaussian exponential correlation function is applied with the proposed method. For the other spatial correlation functions, the proposed method is also suitable. Appendix Table 8 presents the most popular examples of spatial correlation functions. Appendix Table 9 presents the new KMIC spatial correlation functions based on the examples given in Appendix Table 8.

Table 8 Examples of commonly used spatial correlation functions

Spatial correlation functions	Expression	Hyper-parameters θ	Number of hyper-parameters to estimate
Exponential	$\exp\left(-\sum_{k=1}^{d}\theta_k\lvert x_k^{(i)}-x_k^{(j)}\rvert\right)$	$\theta_1, ..., \theta_d$	d
Generalized exponential	$\exp\left(-\sum_{k=1}^{d}\theta_k\lvert x_k^{(i)}-x_k^{(j)}\rvert^{p}\right)$	$\theta_1, ..., \theta_d, p$	$d+1$
Gaussian exponential	$\exp\left(-\sum_{k=1}^{d}\theta_k\lvert x_k^{(i)}-x_k^{(j)}\rvert^{2}\right)$	$\theta_1, ..., \theta_d$	d
Spline	$\prod_{k=1}^{d}\varsigma(\xi_k), \varsigma(\xi_k)=\begin{cases}1-15\xi_k^2+30\xi_k^3, 0\le\xi_k\le 0.2\\ 1.25(1-\xi_k)^3 \quad, 0.2<\xi_k<1\\ 0 \quad, \xi_k\ge 1\end{cases}$, where $\xi_k=\theta_k\lvert x_k^{(i)}-x_k^{(j)}\rvert$	$\theta_1, ..., \theta_d$	d

Table 9 Examples of KMIC spatial correlation functions

Spatial correlation functions	Expression	New hyper-parameters	Number of hyper-parameters to estimate
Exponential	$\exp\left(-\sum_{k=1}^{d}\lambda w_k\lvert x_k^{(i)}-x_k^{(j)}\rvert\right)$	λ	1
Generalized exponential	$\exp\left(-\sum_{k=1}^{d}\lambda w_k\lvert x_k^{(i)}-x_k^{(j)}\rvert^{p}\right)$	λ, p	2
Gaussian exponential	$\exp\left(-\sum_{k=1}^{d}\lambda w_k\lvert x_k^{(i)}-x_k^{(j)}\rvert^{2}\right)$	λ	1
Spline	$\prod_{k=1}^{d}\varsigma(\xi_k), \varsigma(\xi_k)=\begin{cases}1-15\xi_k^2+30\xi_k^3, 0\le\xi_k\le 0.2\\ 1.25(1-\xi_k)^3 \quad, 0.2<\xi_k<1\\ 0 \quad, \xi_k\ge 1\end{cases}$, where $\xi_k=\lambda w_k\lvert x_k^{(i)}-x_k^{(j)}\rvert$	λ	1

References

Albanese D, Filosi M, Visintainer R, Riccadonna S, Jurman G, Furlanello C (2013) Minerva and minepy: a c engine for the mine suite and its r, python and matlab wrappers. Bioinformatics 29(3):407–408

Bouhlel MA, Martins JRRA (2019) Gradient-enhanced kriging for high-dimensional problems. Engineering with Computers 35(1):157–173

Bouhlel MA, Bartoli N, Otsmane A, Morlier J (2016) Improving kriging surrogates of high-dimensional design models by partial least squares dimension reduction. Struct Multidiscip Optim 53(5):935–952

Box GE, Draper NR (1987) Empirical model-building and response surfaces. J R Stat Soc 30(2):229–231

Brest J, Greiner S, Boskovic B, Mernik M, Zumer V (2006) Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. IEEE Trans Evol Comput 10(6):646–657

Buhmann MD (2003) Radial basis functions: theory and implementations, vol 12. Cambridge University Press, Cambridge

Cai X, Qiu H, Gao L, Shao X (2017) Metamodeling for high dimensional design problems by multi-fidelity simulations. Struct Multidiscip Optim 56(1):151–166

Chen L, Qiu H, Gao L, Jiang C, Yang Z (2019) A screening-based gradient-enhanced kriging modeling method for high-dimensional problems. Appl Math Model 69:15–31

Da Veiga S (2015) Global sensitivity analysis with dependence measures. J Stat Comput Simul 85(7):1283–1305

Dong H, Sun S, Song B, Wang P (2019) Multi-surrogate-based global optimization using a score-based infill criterion. Struct Multidiscip Optim 59(2):485–506

Dong H, Song B, Dong Z, Wang P (2018) Scgosr: surrogate-based constrained global optimization using space reduction. Appl Soft Comput 65:462–477

Emmerich MT, Giannakoglou KC, Naujoks B (2006) Single-and multiobjective evolutionary optimization assisted by Gaussian random field metamodels. IEEE Trans Evol Comput 10(4):421–439

Forrester AI, Keane AJ (2009) Recent advances in surrogate-based optimization. Prog Aerosp Sci 45(1–3):50–79

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

Haftka RT, Mroz Z (1986) First-and second-order sensitivity analysis of linear and nonlinearstructures. AIAA J 24(7):1187–1192

Han ZH, Zhang Y, Song CX, Zhang KS (2017) Weighted gradient-enhanced kriging for high-dimensional surrogate modeling and design optimization. AIAA J 55(12):4330–4346

Hartwig L, Bestle D (2017) Compressor blade design for stationary gas turbines using dimension reduced surrogate modeling. Evol Comput

Hemmateenejad B, Baumann K (2018) Screening for linearly and nonlinearly related variables in predictive cheminformatic models. J Chemom 32:e3009

Hollingsworth P, Mavris D (2003) Gaussian process meta-modeling: comparison of Gaussian process training methods. In AIAA's 3rd Annual Aviation Technology, Integration, and Operations (ATIO) Forum (p. 6761)

Jones DR (2001) A taxonomy of global optimization methods based on response surfaces. J Glob Optim 21(4):345–383

Kinney JB, Atwal GS (2014) Equitability, mutual information, and the maximal information coefficient. Proc Natl Acad Sci 111(9):3354–3359

Krige DG (1951) A statistical approach to some basic mine valuation problems on the Witwatersrand. J South Afr Inst Min Metall 52(6):119–139

Kulfan B, Bussoletti J (2006) "Fundamental" parameteric geometry representations for aircraft component shapes. Paper presented at the 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference: The Modeling and Simulation Frontier for Multidisciplinary Design Optimization

Lee K, Cho H, Lee I (2019) Variable selection using Gaussian process regression-based metrics for high-dimensional model approximation with limited data. Struct Multidiscip Optim 59(5):1439–1454

Li C, Wang P, Dong H, Wang X (2018) A simplified shape optimization strategy for blended-wing-body underwater gliders. Struct Multidiscip Optim 58(5):2189–2202

Liu B, Zhang Q, Gielen GG (2014) A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems. IEEE Trans Evol Comput 18(2):180–192

Loeppky JL, Sacks J, Welch WJ (2009) Choosing the sample size of a computer experiment: a practical guide. TECHNOMETRICS 51(4):366–376

Lophaven SN, Nielsen HB, Søndergaard J (2002) Aspects of the matlab toolbox DACE. IMM, Informatics and Mathematical Modelling, The Technical University of Denmark

Matheron G (1963) Principles of geostatistics. Econ Geol 58(8):1246–1266

Mckay MD, Beckman RJ, Conover WJ (1979) Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 21(2):239–245

Michalewicz Z, Schoenauer M (2014) Evolutionary algorithms for constrained parameter optimization problems. Evol Comput 4(1):1–32

Mullur A, Messac A (2005) Extended radial basis functions: more flexible and effective metamodeling. AIAA J 43(6):1306–1315

Powell MJ (1994) A direct search optimization method that models the objective and constraint functions by linear interpolation. In: Advances in optimization and numerical analysis. Springer, Dordrecht, pp 51–67

Rasmussen CE, Williams CKI (2005) Gaussian processes for machine learning. MIT Press, Cambridge

Reshef DN, Reshef YA, Finucane HK, Grossman SR, Mcvean G, Turnbaugh PJ et al (2011) Detecting novel associations in large data sets. Science 334(6062):1518–1524

Sacks J, Welch WJ, Mitchell TJ, Wynn HP (1989) Design and analysis of computer experiments. Stat Sci:409–423

Saltelli A, Ratto M, Andres T, Campolongo F, Cariboni J, Gatelli D et al (2008) Global sensitivity analysis: the primer. Wiley, Hoboken

Sasena MJ (2002) Flexibility and efficiency enhancements for constrained global design optimization with kriging approximations (Doctoral dissertation, University of Michigan)

Schmit LA, Farshi B (1974) Some approximation concepts for structural synthesis. AIAA J 12(5):692–699

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

Smola AJ, Schölkopf B (2004) A tutorial on support vector regression. Stat Comput 14(3):199–222

Sobol IM (2001) Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Math Comput Simul 55(1–3):271–280

Speed T (2011) A correlation for the 21st century. Science 334(6062):1502–1503

Sun GL, Li JB, Dai J, Song ZC, Lang F (2018). Feature selection for IoT based on maximal information coefficient. Futur Gener Comput Syst 89:606–616

Ulaganathan S, Couckuyt I, Dhaene T, Degroote J, Laermans E (2016a) High dimensional kriging metamodelling utilising gradient information. Appl Math Model 40(9–10):5256–5270

Ulaganathan S, Couckuyt I, Dhaene T, Degroote J, Laermans E (2016b) Performance study of gradient-enhanced kriging. Eng Comput 32(1):15–34

Wang H, Jin Y, Doherty J (2017) Committee-based active learning for surrogate-assisted particle swarm optimization of expensive problems. IEEE Trans Cybern 47(9):2664–2677

Wu D, Wang GG (2018) Knowledge assisted optimization for large-scale problems: a review and proposition. In: ASME 2018 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp V02BT03A032–V02BT03A032. American Society of Mechanical Engineers

Wu D, Coatanea E, Wang GG (2017) Dimension reduction and decomposition using causal graph and qualitative analysis for aircraft concept design optimization. In: ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers, New York, pp V02BT03A035–V02BT03A035

Zhao X, Deng W, Shi Y (2013) Feature selection with attributes clustering by maximal information coefficient. Procedia Comput Sci 17(2): 70–79