ICAS 2020 SHANGHAI
International Council of the Aeronautical Sciences
32nd Congress of the International Council of the Aeronautical Sciences
September 6-10, 2021
Pudong Shangri-La, Shanghai, China

IMPROVED WEIGHTED GRADIENT-ENHANCED KRIGING MODEL FOR HIGH-DIMENSIONAL AERODYNAMIC MODELING PROBLEMS

Chen-Zhou Xu, Zhong-Hua Han*, Ke-Shi Zhang & Wen-Ping Song

Institute of Aerodynamic and Multidisciplinary Design Optimization, National Key Laboratory of Science and Technology on Aerodynamic Design and Research, School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072, P. R. China

Abstract

Weighted gradient-enhanced kriging has been demonstrated to be superior to conventional gradient-enhanced kriging when applied to high-dimensional aerodynamic modeling and optimization problems, through the core idea of summing up a series of submodels with much smaller coloration matrices by appropriate weight coefficients. It avoids the prohibit computational cost associated with directly decomposing the large correlation matrix of a gradient-enhanced kriging, and provides a probable way to ameliorate the “curse of dimensionality”. However, with the increase of model training data, the number of required submodels grows rapidly, resulting in another dilemma that the total computation cost of decomposing all the small matrices could become prohibitive. In the paper, an improved formulation of weighted gradient-enhanced kriging is proposed and provides a method to adaptively determine the best suitable number of gradients to be interpolated for each submodel, saving the computation budget for matrix decomposition as much as possible and reaching a win-win situation for both model accuracy and modeling efficiency. Numerical examples are employed to compare the proposed method with other gradient-enhanced modeling approaches. Results demonstrate the advantage of the proposed method in both prediction accuracy and efficiency. It is also applied to the aerodynamic modeling of an RAE2822 airfoil in transonic regime, to further illustrate its capability to support engineering design problems driven by expensive numerical simulations.

Keywords: surrogate model, weighted gradient-enhanced kriging, high-dimensional problem, aerodynamic modeling

1. Introduction

During the past three decades, surrogate-based optimization (SBO)[1]-[4], which makes use of cheap-to-evaluate surrogate models to approximate the output of high-fidelity and expensive numerical simulations, such as computational fluid dynamics (CFD) or finite element analysis (FEA), has attracted much attention in different areas of aerospace engineering[5]-[11] for it is capable of finding a global optimum efficiently[12][13]. As a key role in the optimization mechanism of SBO, surrogate model is referred to as an approximation model of time-consuming analysis code constructed by few sampled data subject to limited computation budget and provides a learning function for the sub-optimization process[14] to obtain new samples towards a global optimum[15]. Currently, the representative surrogate models are polynomial response surface model (PRSM)[16], radial basis function (RBFs)[17], Kriging[18][19], artificial neural network (ANN)[20][21], support vector regression (SVR)[22], multivariate interpolation and regression (MIR)[23], polynomial chaos expansion (PCE)[24], etc. Among them, kriging model, also known as Gaussian process regressions[25], has gained popularity in the field of aerodynamic modeling and design optimization[26]-[28], structural modeling[29], and multidisciplinary design optimization (MDO)[4][7][30] because of its high fitting capability for nonlinear and multimodal functions, along with a unique feature of providing mean-squared-error estimation[14]. However, as most engineering problems are pursuing higher-fidelity and more global

optimal design nowadays, the number of design variables is increasing rapidly, and more strong constraints need to be taken into consideration simultaneously, which poses a great challenge to the prohibitive computational cost of building sufficiently accurate surrogate models for these high-dimensional problems[4]. To tackle or at least ameliorate this so-called "curse of dimensionality" [31][32], the continuous effort has been devoted into surrogate-based modeling and optimization algorithm during the past two decades, such as integrating surrogates with high dimensional model representation (HDMR)[33][34] or exploring auxiliary information to enhance the surrogates[35].

In a high-dimensional problem, the surrogate model built through initial samples could not be accurate enough to approximate the complex landscape of the objective function so that the newly added sample sequence converges slowly, which results in calling more expensive numerical simulations to enhance the prediction accuracy and find the real optimal solution in the whole design space as well. To solve this problem, surrogate models with cheap auxiliary information to improve model accuracy have been developed recently and two kinds of methods have been concerned. One is the variable-fidelity model (VFM)[36]-[39] that uses lower-fidelity, cheaper analysis model to assist the prediction of a high-fidelity, expensive analysis model, and the other is the gradient-enhanced surrogate model[40][41] with the cheap gradients computed by an adjoint method[42]. Here we are mainly concerned with the ordinary kriging model enhanced by cheap gradient information, and the issue related to the usage of lower-fidelity data is beyond the scope of this paper.

The inspiration for interpolating cheap gradients in the construction of a kriging to dramatically enhance the model accuracy was firstly proposed by Morris[43] in 1993. It was then introduced to aerodynamic configuration optimization of a low-boom supersonic business jet by Chung and Alonso[40] in 2002, with the cheap gradients obtained by adjoint method and used to augment the training data. Since the big potential of using an adjoint approach to compute the gradient efficiently, where the cost is almost independent of the number of input variables, surrogate enhanced with gradients has largely inspirited the research and development in the field of engineering design optimization. A systemic overview of the gradient-enhanced surrogate model can be found in Ref.[44].

According to the ways of incorporating gradients into the construction of a kriging, gradient-enhanced kriging (GEK) can be classified into two categories: indirect-GEK[45]-[47] and direct-GEK[41]. The methodology of indirect-GEK is that the gradients are reverted to additional function values close to the sampling sites via first-order Taylor's expansion[48]. The training data set is then augmented by these extra function values and the model fitting process could be easy to implement, sharing the same mathematical formulation of a conventional kriging. The main drawback of this approach is that its approximate accuracy heavily depends on the step size of the Taylor's expansion, which might introduce numerical errors with large step size or make model correlation matrix ill-conditioned when the step size is too small. Besides, along with the advancement of the optimization process, the newly added samples will cluster around the current optimal, leading to a rapid growth in the condition number of the correlation matrix as well. On the contrary, the gradients in a direct-GEK are recognized as part of model training data to avoid this dilemma, with an assumption that the predictor is defined by augmenting the weighted sum of the functional values with the weighted sum of the gradients[49]. Taking all the extra correlations between the function values and the gradients as well as the correlations between the gradients themselves into consideration, the correlation matrix of a direct-GEK could easily become enormous but make the fitted model more accurate and robust. Laurenceau[50] compared the accuracy of a kriging, indirect-GEK and direct-GEK model for aerodynamic data prediction, and found that the two GEK models were more accurate than the kriging model with the same training data set. Zimmermann[51] demonstrated that the condition number of correlation matrix of an indirect-GEK is larger than that of a direct-GEK in most cases through theoretical analysis. In addition, some numerical examples by Laurent[44] showed that an indirect-GEK shares similar model accuracy to a direct-GEK in low-dimensional problems, but gets worse in high-dimensional problems. Here we are only concerned with the improvement in direct gradient-enhanced ordinary kriging, and the term GEK in the remaining parts of the article refers to this

approach. Some improved GEK model using universal trend function to enhance the accuracy is also beyond the scope of this paper.

Although GEK has been successfully applied to many low- or medium-dimensional problems during the past few years, it is still suffering from the prohibitive computational cost when applied to higher-dimensional optimization problems. As mentioned before, the introduce of additional cheap gradients helps enhance the model accuracy of a kriging significantly while training such a GEK model could be extremely costly in high-dimensional problems as the correlation matrix is largely expanded when the gradients are incorporated, sizing from $n \times n$ to $(n+nm)\times(n+nm)$, where n is the number of sampling sites, and m is the number of dimensions (or design variables). The curse of dimensionality is still a "bottle-neck" that limits the further development and applications of a GEK, which motivates many researchers to work on cutting down the modeling cost. Bouhlel[47] applied the partial least squares (PLS) method on each sampling site and selected the most relevant approximating points to include in the correlation matrix given by the PLS information, to control the size of the correlation matrix and reduce the number of hyperparameters. Chen[35] evaluated the influence of each input variable on the final prediction results, and utilized an empirical evaluation rule, where a trade-off between the model accuracy and modeling efficiency can be achieved, to partially introduce the gradients. Then, a partial gradient-enhanced kriging (PGEK) was established to alleviate the curse of dimensionality. Han[49] proposed a weighted gradient-enhanced kriging (WGEK) model, which can dramatically improve the model fitting efficiency for high dimensional problems. Its core idea was to build a series of submodels with much smaller correlation matrices and then sum them up with appropriate weight coefficients. Each submodel was built through all the observed function values at all sampling sites and the gradients at one site only and thus the number of submodels was equal to that of total sampling sites. However, WGEK is still suffering from the prohibitive computational cost especially when there exist numerous samples. For each existing sample point, it is necessary to build a submodel and the increasing of the number of submodels leads to a decline in the total modeling fitting efficiency. Apart from this, the divided group of sampling sites buckles the correlations between the gradients. That is to say, the second-order cross-partial derivative terms are removed from the correlation matrix, which sacrifices the model accuracy as a price for increasing modeling efficiency.

The main objective of this paper is to improve the current WGEK, and develop a novel formulation of a gradient-enhanced kriging model called adaptive weighted gradient-enhanced kriging (AWGEK), which can improve both the model accuracy and fitting efficiency of a gradient-enhanced surrogate. The proposed AWGEK provides a mechanism to adaptively determine the group of sampling sites according to the dimension of the problem and the current number of sampling sites, interpolating the gradients at more than one site in each submodel. It overcomes the main drawback of a WGEK that there would exist numerous submodels corresponding to each sample in high-dimensional modeling problems, and adjusts grouping strategy of gradients via an evaluation rule, which adaptively minimizes the total cost of correlation matrix decomposition.

This paper continues in section 2 for the description of the proposed AWGEK model and analytical test cases are to be used to validate the correctness in Section 3. In Section 4, the proposed method is demonstrated by the aerodynamic modeling of an RAE2822 airfoil in transonic regime, with 36 design variables. At last, general conclusions will be drawn and the future work will be discussed.

2. Formulation of Adaptive Weighted Gradient-Enhanced Kriging Model

2.1 Concept of Weighted Gradient-Enhanced Kriging

As mentioned in the introduction, GEK can significantly improve the efficiency of an optimization with the assistance of auxiliary gradient information to enhance the model accuracy. Nevertheless, for high dimensional problems, if existing numerous sample points in the model training data set, the scale of correlation matrix $(n+nm)\times(n+nm)$ would readily become huge, causing difficulties with matrix decomposition and inversion. From the perspective of saving the large cost of the correlation

matrix decomposition, Han[49] proposed a novel weighted gradient-enhanced kriging to build a series of submodels with smaller correlation matrices and then to sum them up with appropriate weight coefficients. Each submodel is built through all the observed function values while interpolating the gradients at one sampling site only, which means there will be n submodels in total. Therefore, the training data set $(\mathbf{S}_i, \mathbf{y}_{\mathrm{S},i})$ for the i-th submodel changes to:

$$\begin{aligned} \mathbf{S}_i &= [\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(n)}, \underbrace{\mathbf{x}^{(i)}, \cdots, \mathbf{x}^{(i)}}_{m}]^{\mathrm{T}} \in \mathbb{R}^{(n+m)\times m}, \\ \mathbf{y}_{\mathrm{S},i} &= [y^{(1)}, \cdots, y^{(n)}, \frac{\partial y^{(i)}}{\partial x_1}, \cdots, \frac{\partial y^{(i)}}{\partial x_m}]^{\mathrm{T}} \in \mathbb{R}^{n+m}. \quad i = 1, \cdots, n \end{aligned} \tag{1}$$

Following the detailed derivation in Ref.[49], we can obtain the predictor $\hat{y}_i(\mathbf{x})$ of the i-th submodel in form of:

$$\hat{y}_i(\mathbf{x}) = \beta_0 + \overline{\mathbf{r}}_i^{\mathrm{T}} \underbrace{\overline{\mathbf{R}}_i^{-1}\left(\mathbf{y}_{\mathrm{S},i} - \beta_0 \overline{\mathbf{F}}_i\right)}_{\mathbf{v}_{GEK,i}}, \tag{2}$$

where

$$\begin{aligned} \beta_0 &= \left(\overline{\mathbf{F}}_i^{\mathrm{T}} \overline{\mathbf{R}}_i^{-1} \overline{\mathbf{F}}_i\right)^{-1} \overline{\mathbf{F}}_i^{\mathrm{T}} \overline{\mathbf{R}}_i^{-1} \mathbf{y}_{\mathrm{S},i}, \\ \overline{\mathbf{F}}_i &= (\underbrace{1, \cdots, 1}_{n}, \underbrace{0, \cdots, 0}_{m})^{\mathrm{T}} \in \mathbb{R}^{n+m}, \\ \overline{\mathbf{R}}_i &= \begin{bmatrix} \mathbf{R} & \partial \mathbf{R}_i \\ \partial \mathbf{R}_i^{\mathrm{T}} & \partial^2 \mathbf{R}_i \end{bmatrix} \in \mathbb{R}^{(n+m)\times(n+m)}, \\ \overline{\mathbf{r}}_i &= \begin{bmatrix} \mathbf{r} \\ \partial \mathbf{r}_i \end{bmatrix} \in \mathbb{R}^{n+m}. \end{aligned} \tag{3}$$

To be more specific, the correlation matrix $\overline{\mathbf{R}}_i$ and correlation vector $\overline{\mathbf{r}}_i$ for the i-th submodel are given by

$$\begin{aligned} \mathbf{R} &= \begin{bmatrix} R(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}) & \cdots & R(\mathbf{x}^{(1)}, \mathbf{x}^{(n)}) \\ \vdots & \ddots & \vdots \\ R(\mathbf{x}^{(n)}, \mathbf{x}^{(1)}) & \cdots & R(\mathbf{x}^{(n)}, \mathbf{x}^{(n)}) \end{bmatrix} \in \mathbb{R}^{n\times n}, \\ \partial \mathbf{R}_i &= \begin{bmatrix} \frac{\partial R(\mathbf{x}^{(1)}, \mathbf{x}^{(i)})}{\partial x_1^{(i)}} & \cdots & \frac{\partial R(\mathbf{x}^{(1)}, \mathbf{x}^{(i)})}{\partial x_m^{(i)}} \\ \vdots & \ddots & \vdots \\ \frac{\partial R(\mathbf{x}^{(n)}, \mathbf{x}^{(i)})}{\partial x_1^{(i)}} & \cdots & \frac{\partial R(\mathbf{x}^{(n)}, \mathbf{x}^{(i)})}{\partial x_m^{(i)}} \end{bmatrix} \in \mathbb{R}^{n\times m}, \\ \partial^2 \mathbf{R}_i &= \begin{bmatrix} \frac{\partial^2 R(\mathbf{x}^{(i)}, \mathbf{x}^{(i)})}{\partial^2 x_1^{(i)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(i)}, \mathbf{x}^{(i)})}{\partial x_1^{(i)} \partial x_m^{(i)}} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 R(\mathbf{x}^{(i)}, \mathbf{x}^{(i)})}{\partial x_m^{(i)} \partial x_1^{(i)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(i)}, \mathbf{x}^{(i)})}{\partial^2 x_m^{(i)}} \end{bmatrix} \in \mathbb{R}^{m\times m}, \end{aligned} \qquad \begin{aligned} \mathbf{r} &= \left[R(\mathbf{x}^{(1)}, \mathbf{x}), \cdots, R(\mathbf{x}^{(n)}, \mathbf{x})\right]^{\mathrm{T}} \in \mathbb{R}^{n}, \\ \partial \mathbf{r}_i &= \left[\frac{\partial R(\mathbf{x}^{(i)}, \mathbf{x})}{\partial x_1^{(i)}}, \cdots, \frac{\partial R(\mathbf{x}^{(i)}, \mathbf{x})}{\partial x_m^{(i)}}\right]^{\mathrm{T}} \in \mathbb{R}^{m}, \end{aligned} \tag{4}$$

respectively. In addition, the MSE of the i-th submodel is given by

$$MSE_i\left[\hat{y}_i(\mathbf{x})\right] = s_i^2(\mathbf{x}) = \sigma^2 \left[1.0 - \overline{\mathbf{r}}_i^{\mathrm{T}} \overline{\mathbf{R}}_i^{-1} \overline{\mathbf{r}}_i + \left(1 - \overline{\mathbf{F}}_i^{\mathrm{T}} \overline{\mathbf{R}}_i^{-1} \overline{\mathbf{r}}_i\right)^2 \Big/ \left(\overline{\mathbf{F}}_i^{\mathrm{T}} \overline{\mathbf{R}}_i^{-1} \overline{\mathbf{F}}_i\right)\right]. \tag{5}$$

Once n sub-GEK models are constructed by following these steps, an extra weight coefficient kriging (WCK) is then employed to offer propriate weight coefficients for summing all the submodels up, in which the prediction at any untried $\mathbf{x}$ is defined by

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{n} \omega_i \hat{y}_i(\mathbf{x}), \tag{6}$$

through the data set $\left(\mathbf{S},\hat{\mathbf{y}}_{\mathrm{S,sub}}\right)$ given by

$$\begin{aligned}\mathbf{S}&=\left[\mathbf{x}^{(1)},\cdots,\mathbf{x}^{(n)}\right]^{\mathrm{T}}\in\mathbb{R}^{n\times m},\\ \hat{\mathbf{y}}_{\mathbf{S},sub}(\mathbf{x})&=\left[\hat{y}_1(\mathbf{x}),\cdots,\hat{y}_n(\mathbf{x})\right]^{\mathrm{T}}\in\mathbb{R}^{n}.\end{aligned}\tag{7}$$

Note that $\hat{y}_i(\mathbf{x})$ represents the prediction of the i-th submodel and is always changing with the location of the untried $\mathbf{x}$. After determining these submodels and weight coefficients, the resulting WGEK predictor is of the form[49]:

$$\hat{y}(\mathbf{x})=\beta_0+\mathbf{r}^{\mathrm{T}}(\mathbf{x})\mathbf{R}^{-1}\left[\hat{\mathbf{y}}_{\mathbf{S},sub}(\mathbf{x})-\beta_0\mathbf{F}\right],\tag{8}$$

and its MSE is given by

$$\begin{aligned}MSE\left[\hat{y}(\mathbf{x})\right]=s^2(\mathbf{x})=\mathrm{var}\left(\sum_{i=1}^{n}w_i\hat{Y}_i\right)=\sum_{i=1}^{n}w_i^{\,2}\,\mathrm{var}\left(\hat{Y}_i\right)\\ +\sum_{i,j=1,i\neq j}^{n}2w_iw_j\,\mathrm{cov}\left(\hat{Y}_i,\hat{Y}_j\right)=\left(\sum_{i=1}^{n}w_is_i\right)^2.\end{aligned}\tag{9}$$

Due to the fact that each submodel only interpolate gradients at one site, the correlation matrix size is as small as $(n+m)\times(n+m)$. The floating-point operations per second (FLOPs) of decomposing such a matrix by Cholesky method is around the third power of matrix size $O((n+m)^3)$. As the number of submodels is equal to that of evaluated sample points, the total FLOPs of decomposing all the matrices for training a WGEK is $O(n(n+m)^3)$. In contrast, the required FLOPs of decomposing a GEK correlation matrix is around $O(n^3(1+m)^3)$. As a result, training a WGEK is about $n^2(1+m)^3/(n+m)^3$ times faster than training a GEK. However, the basic theoretical assumption of WGEK model not only cuts off the interactions between gradient information in each submodel as second-order cross derivatives are eliminated from the correlation matrix, leading to a decrease on model accuracy, but also limits the further improvement of its modeling efficiency especially when existing numerous sample points, which means the total cost of training these submodels could become unbearable in high-dimensional problems. This motivates the research of this paper.

2.2 Adaptive Grouping Mechanism of Gradients for Submodels

The core idea of AWGEK, like that of WGEK, is to build a series of sub-GEK models and sum them up with appropriate weight coefficients as well. The only difference is that the number of gradients introduced to each submodel is determined by an adaptive grouping mechanism, no longer restricted to considering gradients at one sampling site only in each submodel. Therefore, the size of correlation matrix could slightly increase since gradients of more than one site might be contained in a submodel, but the total number of need-to-build submodels decreases. Note that the more gradients are interpolated in each submodel, the fewer submodels we need in training an AWGEK, but we must always keep in mind that the correlation matrix size of a submodel shouldn't be too large. There hence exists a most suitable strategy to divide the gradients at all sampling sites into several groups accounting for each submodel, which makes the total cost of training all submodels the lowest.

Assuming a more general formulation of weighted gradient-enhanced kriging model, the number of gradients at evaluated sampling sites included in the i-th sub-model are defined as a_i, and k submodels are required in total to cover gradients of all the sample points. That is to say, it has to satisfy the following equality constraint:

$$\sum_{i=1}^{k}a_i=n,\tag{10}$$

where a_i, k are both integers, and n denotes the number of sample points utilized for training an AWGEK. Then, the correlation matrix size of the i-th submodel will be $(n+a_im)\times(n+a_im)$ and the

FLOPs of decomposing this matrix by Cholesky method is around $O((n+a_i m)^3)$. As a result, the total FLOPs of decomposing all k matrices is given by

$$F(k,\mathbf{a}) = \left(n+a_1 m\right)^3 + \left(n+a_2 m\right)^3 + ... + \left(n+a_k m\right)^3 = \sum_{i=1}^{k} (n+a_i m)^3, \tag{11}$$

where $\mathbf{a} = [a_1,...,a_k]^{\mathrm{T}}$. Here, we aim to find a pair of $\mathbf{a}$ and k, which minimizes the total computational cost $F(k,\mathbf{a})$ subject to the constraint. To solve this optimization problem varying with discrete integer variables, a two-step strategy is adopted, where the best grouping vector $\mathbf{a}$ is first obtained under the assumed known k and we search for the optimal k afterwards.

Step I. Determine the best grouping vector $\mathbf{a}$ with an assumed known k. Before deriving the best grouping plan with an arbitrary value of k ($k \geq 2$ and k is an integer), we might firstly consider a specific situation with k = 2 and a smaller subset of n for convenience. Note that the minimization problem could become:

$$\begin{aligned} &\min. \quad F_{subset} = \left(n+a_1 m\right)^3 + \left(n+a_2 m\right)^3, \\ &w.r.t. \quad a_1, a_2 \in N_+, \\ &s.t. \quad\ a_1 + a_2 = n_{subset}, \end{aligned} \tag{12}$$

where $n_{subset} \subseteq n$.

Solving the above objective function with an equality constraint by Lagrange Multiplier method, a_i can be found by solving the following linear equations

$$\begin{cases} \dfrac{\partial L}{\partial a_1} = 3m\left(n+a_1 m\right)^2 + \mu = 0, \\ \dfrac{\partial L}{\partial a_2} = 3m\left(n+a_2 m\right)^2 + \mu = 0, \\ \dfrac{\partial L}{\partial \mu} = a_1 + a_2 - n_{subset} = 0, \end{cases} \tag{13}$$

where $L = F_{subset} + \mu\left(a_1 + a_2 - n_{subset}\right)$ and μ is the Lagrange multiplier used to change the constrained minimization of F_{subset} into an unconstrained one. Theoretically, we could easily obtain the optimum $a_1 = a_2 = n_{subset}/2$ if n_{subset} is an even number. Consider the situation that n_{subset} is odd, and then we suppose instead

$$a_1 = \frac{n_{subset}-1}{2} - \varepsilon, \quad a_2 = \frac{n_{subset}+1}{2} + \varepsilon, \tag{14}$$

(29)

where ε is a natural number. Substituting eq.(14) in eq.(12) leads to the following formulation of F_{subset}:

$$\begin{aligned} F_{subset} &= 2n^3 + 3n^2 m n_{subset} + 3nm^2(a_1^2 + a_2^2) + m^3(a_1^3 + a_2^3) \\ &= 2n^3 + 3n^2 m n_{subset} + (3nm^2 + m^3 n_{subset})(a_1^2 + a_2^2) - m^3 a_1 a_2 n_{subset} \\ &= F_{const} + (3nm^2 + m^3 n_{subset}) \cdot 2(\varepsilon + \varepsilon^2) + m^3 n_{subset}(\varepsilon + \varepsilon^2) \\ &= F_{const} + (6nm^2 + 3m^3 n_{subset}) \cdot (\varepsilon + \varepsilon^2) \end{aligned} \tag{15}$$

where F_{const} is the constant terms independent of ε. Note that ε is a natural number and is supposed to be as small as possible in order to reach the extreme value point of F_{subset}. Hence, when ε is set to zero with an odd n_{subset}, F_{subset} is the smallest with respect to the integer grouping vector $\mathbf{a} = \left[(n_{subset}-1)/2, (n_{subset}+1)/2\right]$. In other words, whether n_{subset} is odd or even, the closer the two values a_1, a_2 are, the smaller the total computational cost F_{subset} is. Since the chosen group a_1, a_2 and n_{subset}

are all arbitrary, the deduction could be easily generalized to the situation when $k>2$. Therefore, it comes to the conclusion that the closer each component in the grouping vector $\mathbf{a}$ under any assumed known k is, the smaller the computational cost F is.

Step II. Perform a loop for searching the optimal *k*. As we have obtained the best self-adaptive grouping strategy with the gradients distributed averagely in each sub-GEK model once the value of k is given, a cheap loop is adopted here to search for the optimal k. We compare the computational cost $F(k)$ with varying k value from 1 to n and determine the final value with the smallest total FLOPs. Note that AWGEK is a more generalized description of gradient-enhanced kriging model, for it equates to GEK when k = 1 and becomes WGEK when k = n.

2.3 Novel Formulation of Proposed AWGEK

To build an AWGEK, once the grouping strategy of gradients ought to be interpolated in each submodel is determined, the following construction steps are almost the same as that of building a WGEK. For an m-dimensional problem, suppose that n spatial sites are sampled and evaluated by running expensive analysis solvers to get the responses as well as their gradients with respect to all the design variables. As n and m is known and fixed in the whole process of training a surrogate model, the number of required submodels are automatically calculated through the proposed method in the previous section. and we still denote it as k. As a result, the prediction of an AWGEK at any untried $\mathbf{x}$ is defined by:

$$\hat{y}(\mathbf{x})=\sum_{i=1}^{k}\omega_i\hat{y}_i(\mathbf{x}). \tag{16}$$

For the i-th submodel, the gradients at a_i sampling sites are contained in its training date sets $(\mathbf{S}_i,\mathbf{y}_{\mathrm{S},i})$, which are collected as:

$$\begin{aligned}
\mathbf{S}_i &=[\mathbf{x}^{(1)},\cdots,\mathbf{x}^{(n)},\underbrace{\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\cdots,\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\cdots,\mathbf{x}^{(\sum_{j=1}^{i}a_j)},\cdots,\mathbf{x}^{(\sum_{j=1}^{i}a_j)}}_{a_i\cdot m}]^{\mathrm{T}}\in\mathbb{R}^{(n+a_i\cdot m)\times m},\\
\mathbf{y}_{\mathbf{S},i} &=[y^{(1)},\cdots,y^{(n)},\frac{\partial y^{(\sum_{j=1}^{i}a_j-a_i+1)}}{\partial x_1},\cdots,\frac{\partial y^{(\sum_{j=1}^{i}a_j-a_i+1)}}{\partial x_m},\cdots,\frac{\partial y^{(\sum_{j=1}^{i}a_j)}}{\partial x_1},\cdots,\frac{\partial y^{(\sum_{j=1}^{i}a_j-a_i)}}{\partial x_m}]^{\mathrm{T}}\in\mathbb{R}^{n+a_i\cdot m}.
\end{aligned} \tag{17}$$

Note that we merely choose a_i sampling sites in sequence from the whole training data set. That is the say, the first a_1 sampling sites are chosen for the first submodel and the next a_2 sampling sites (with the index of $a_1+1,a_1+2,...,a_1+a_2$) are chosen for the second submodel. However, the impact of selecting different sampling sites for each submodel is ignored here, which could be the future work to improve this method. Following the similar derivation in section 2.1, we can also obtain the same predictor $\hat{y}_i(\mathbf{x})$ of the i-th submodel in form of

$$\hat{y}_i(\mathbf{x})=\beta_0+\overline{\mathbf{r}}_i^{\mathrm{T}}\underbrace{\overline{\mathbf{R}}_i^{-1}\left(\mathbf{y}_{\mathbf{S},i}-\beta_0\overline{\mathbf{F}}_i\right)}_{\mathbf{v}_{GEK,i}}, \tag{18}$$

where

$$\begin{aligned}
\beta_0 &=\left(\overline{\mathbf{F}}_i^{\mathrm{T}}\overline{\mathbf{R}}_i^{-1}\overline{\mathbf{F}}_i\right)^{-1}\overline{\mathbf{F}}_i^{\mathrm{T}}\overline{\mathbf{R}}_i^{-1}\mathbf{y}_{\mathbf{S},i},\\
\overline{\mathbf{F}}_i &=(\underbrace{1,\cdots,1}_{n},\underbrace{0,\cdots,0}_{a_i\cdot m})^{\mathrm{T}}\in\mathbb{R}^{n+a_i\cdot m},\\
\overline{\mathbf{R}}_i &=\begin{bmatrix}\mathbf{R} & \partial\mathbf{R}_i\\ \partial\mathbf{R}_i^{\mathrm{T}} & \partial^2\mathbf{R}_i\end{bmatrix}\in\mathbb{R}^{(n+a_i\cdot m)\times(n+a_i\cdot m)},\\
\overline{\mathbf{r}}_i &=\begin{bmatrix}\mathbf{r}\\ \partial\mathbf{r}_i\end{bmatrix}\in\mathbb{R}^{n+a_i\cdot m}.
\end{aligned} \tag{19}$$

Since each submodel contains the gradients at more than one sampling site, the correlation matrix $\overline{\mathbf{R}}_i$ and correlation vector $\overline{\mathbf{r}}_i$ for the i-th submodel are changed to

$$\mathbf{R} = \begin{bmatrix} R(\mathbf{x}^{(1)},\mathbf{x}^{(1)}) & \cdots & R(\mathbf{x}^{(1)},\mathbf{x}^{(n)}) \\ \vdots & \ddots & \vdots \\ R(\mathbf{x}^{(n)},\mathbf{x}^{(1)}) & \cdots & R(\mathbf{x}^{(n)},\mathbf{x}^{(n)}) \end{bmatrix} \in \mathbb{R}^{n\times n},$$

$$\partial\mathbf{R}_i = \begin{bmatrix} \frac{\partial R(\mathbf{x}^{(1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial x_1^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial R(\mathbf{x}^{(1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial x_m^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial R(\mathbf{x}^{(1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial x_1^{(\sum_{j=1}^{i}a_j)}} & \cdots & \frac{\partial R(\mathbf{x}^{(1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial x_m^{(\sum_{j=1}^{i}a_j)}} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \frac{\partial R(\mathbf{x}^{(n)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial x_1^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial R(\mathbf{x}^{(n)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial x_m^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial R(\mathbf{x}^{(n)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial x_1^{(\sum_{j=1}^{i}a_j)}} & \cdots & \frac{\partial R(\mathbf{x}^{(n)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial x_m^{(\sum_{j=1}^{i}a_j)}} \end{bmatrix} \in \mathbb{R}^{n\times a_i\cdot m}, \tag{20}$$

$$\partial^2\mathbf{R}_i = \begin{bmatrix} \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial^2 x_1^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial x_1^{(\sum_{j=1}^{i}a_j-a_i+1)}\partial x_m^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial x_1^{(\sum_{j=1}^{i}a_j-a_i+1)}\partial x_1^{(\sum_{j=1}^{i}a_j)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial x_1^{(\sum_{j=1}^{i}a_j-a_i+1)}\partial x_m^{(\sum_{j=1}^{i}a_j)}} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial x_m^{(\sum_{j=1}^{i}a_j-a_i+1)}\partial x_1^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial^2 x_m^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial x_m^{(\sum_{j=1}^{i}a_j-a_i+1)}\partial x_1^{(\sum_{j=1}^{i}a_j)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial x_m^{(\sum_{j=1}^{i}a_j-a_i+1)}\partial x_m^{(\sum_{j=1}^{i}a_j)}} \\ \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial x_1^{(\sum_{j=1}^{i}a_j)}\partial x_1^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial x_m^{(\sum_{j=1}^{i}a_j)}\partial x_1^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial^2 x_1^{(\sum_{j=1}^{i}a_j)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial x_1^{(\sum_{j=1}^{i}a_j)}\partial x_m^{(\sum_{j=1}^{i}a_j)}} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial x_m^{(\sum_{j=1}^{i}a_j)}\partial x_1^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j)},\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)})}{\partial x_m^{(\sum_{j=1}^{i}a_j)}\partial x_m^{(\sum_{j=1}^{i}a_j-a_i+1)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial x_m^{(\sum_{j=1}^{i}a_j)}\partial x_1^{(\sum_{j=1}^{i}a_j)}} & \cdots & \frac{\partial^2 R(\mathbf{x}^{(\sum_{j=1}^{i}a_j)},\mathbf{x}^{(\sum_{j=1}^{i}a_j)})}{\partial^2 x_m^{(\sum_{j=1}^{i}a_j)}} \end{bmatrix} \in \mathbb{R}^{a_i\cdot m\times a_i\cdot m},$$

and

$$\begin{aligned} \mathbf{r} &= \left[R(\mathbf{x}^{(1)},\mathbf{x}),\cdots,R(\mathbf{x}^{(n)},\mathbf{x})\right]^{\mathrm{T}} \in \mathbb{R}^{n}, \\ \partial\mathbf{r}_i &= \left[\frac{\partial R(\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\mathbf{x})}{\partial x_1^{(\sum_{j=1}^{i}a_j-a_i+1)}},\cdots,\frac{\partial R(\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\mathbf{x})}{\partial x_m^{(\sum_{j=1}^{i}a_j-a_i+1)}},\cdots,\frac{\partial R(\mathbf{x}^{(\sum_{j=1}^{i}a_j)},\mathbf{x})}{\partial x_1^{(\sum_{j=1}^{i}a_j)}},\cdots,\frac{\partial R(\mathbf{x}^{(\sum_{j=1}^{i}a_j)},\mathbf{x})}{\partial x_m^{(\sum_{j=1}^{i}a_j)}}\right]^{\mathrm{T}} \in \mathbb{R}^{a_i\cdot m}, \end{aligned} \tag{21}$$

respectively. Also, the MSE of the i-th submodel is given by

$$MSE_i\left[\hat{y}_i(\mathbf{x})\right] = s_i^2(\mathbf{x}) = \sigma^2\left[1.0-\overline{\mathbf{r}}_i^{\mathrm{T}}\overline{\mathbf{R}}_i^{-1}\overline{\mathbf{r}}_i + \left(1-\overline{\mathbf{F}}_i^{\mathrm{T}}\overline{\mathbf{R}}_i^{-1}\overline{\mathbf{r}}_i\right)^2 \Big/ \left(\overline{\mathbf{F}}_i^{\mathrm{T}}\overline{\mathbf{R}}_i^{-1}\overline{\mathbf{F}}_i\right)\right],\ i=1,2,...,k \tag{22}$$

Once a total of k submodels are built following the above steps, a couple of propriate weight coefficients are required for summing all the submodels up. From the definition of the submodel in the previous section, we could obviously observe that the i-th submodel not only needs to pass through all n samples precisely, but also requires to be tangent to the real-world function values at $[\underbrace{\mathbf{x}^{(\sum_{j=1}^{i}a_j-a_i+1)},\cdots,\mathbf{x}^{(\sum_{j=1}^{i}a_j)}}_{a_i}]^{\mathrm{T}}$ sampling sites. In other words, the weight coefficients of these k submodels must satisfied the following interpolation conditions:

$$
\begin{cases}
\forall \mathbf{x}, \quad w_1 + w_2 + \ldots + w_k = 1, \\
w_1\left(\mathbf{x}^{(1)}\right) = 1, w_2\left(\mathbf{x}^{(1)}\right) = 0, \ldots, w_k\left(\mathbf{x}^{(1)}\right) = 0 \\
\quad \vdots \\
w_1\left(\mathbf{x}^{(a_1)}\right) = 1, w_2\left(\mathbf{x}^{(a_1)}\right) = 0, \ldots, w_k\left(\mathbf{x}^{(a_1)}\right) = 0 \\
\quad \vdots \\
w_1\left(\mathbf{x}^{\left(\sum_{j=1}^{i} a_j - a_i + 1\right)}\right) = 0, w_2\left(\mathbf{x}^{\left(\sum_{j=1}^{i} a_j - a_i + 1\right)}\right) = 0, \ldots, w_i\left(\mathbf{x}^{\left(\sum_{j=1}^{i} a_j - a_i + 1\right)}\right) = 1, \ldots, w_k\left(\mathbf{x}^{\left(\sum_{j=1}^{i} a_j - a_i + 1\right)}\right) = 0 \\
\quad \vdots \\
w_1\left(\mathbf{x}^{\left(\sum_{j=1}^{i} a_j\right)}\right) = 0, w_2\left(\mathbf{x}^{\left(\sum_{j=1}^{i} a_j\right)}\right) = 0, \ldots, w_i\left(\mathbf{x}^{\left(\sum_{j=1}^{i} a_j\right)}\right) = 1, \ldots, w_k\left(\mathbf{x}^{\left(\sum_{j=1}^{i} a_j\right)}\right) = 0 \\
\quad \vdots \\
w_1\left(\mathbf{x}^{(n-a_k+1)}\right) = 0, w_2\left(\mathbf{x}^{(n-a_k+1)}\right) = 0, \ldots, w_k\left(\mathbf{x}^{(n-a_k+1)}\right) = 1 \\
\quad \vdots \\
w_1\left(\mathbf{x}^{(n)}\right) = 0, w_2\left(\mathbf{x}^{(n)}\right) = 0, \ldots, w_k\left(\mathbf{x}^{(n)}\right) = 1
\end{cases}
\tag{23}
$$

and tangency conditions:

$$
\begin{cases}
\left.\frac{\partial(w_1\hat{y}_1)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(1)}} = \frac{\partial y^{(1)}}{\partial x_d}, \left.\frac{\partial(w_2\hat{y}_2)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(1)}} = 0.0, \ldots, \left.\frac{\partial(w_k\hat{y}_k)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(1)}} = 0.0 \\
\quad \vdots \\
\left.\frac{\partial(w_1\hat{y}_1)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(a_1)}} = \frac{\partial y^{(a_1)}}{\partial x_d}, \left.\frac{\partial(w_2\hat{y}_2)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(a_1)}} = 0.0, \ldots, \left.\frac{\partial(w_k\hat{y}_k)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(a_1)}} = 0.0 \\
\quad \vdots \\
\left.\frac{\partial(w_1\hat{y}_1)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{\left(\sum_{j=1}^{i} a_j - a_i + 1\right)}} = 0.0, \left.\frac{\partial(w_2\hat{y}_2)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{\left(\sum_{j=1}^{i} a_j - a_i + 1\right)}} = 0.0, \ldots, \left.\frac{\partial(w_i\hat{y}_i)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{\left(\sum_{j=1}^{i} a_j - a_i + 1\right)}} = \frac{\partial y^{\left(\sum_{j=1}^{i} a_j - a_i + 1\right)}}{\partial x_d}, \ldots, \left.\frac{\partial(w_k\hat{y}_k)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{\left(\sum_{j=1}^{i} a_j - a_i + 1\right)}} = 0.0 \\
\quad \vdots \\
\left.\frac{\partial(w_1\hat{y}_1)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{\left(\sum_{j=1}^{i} a_j\right)}} = 0.0, \left.\frac{\partial(w_2\hat{y}_2)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{\left(\sum_{j=1}^{i} a_j\right)}} = 0.0, \ldots, \left.\frac{\partial(w_i\hat{y}_i)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{\left(\sum_{j=1}^{i} a_j\right)}} = \frac{\partial y^{\left(\sum_{j=1}^{i} a_j\right)}}{\partial x_d}, \ldots, \left.\frac{\partial(w_k\hat{y}_k)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{\left(\sum_{j=1}^{i} a_j\right)}} = 0.0 \\
\quad \vdots \\
\left.\frac{\partial(w_1\hat{y}_1)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(n-a_k+1)}} = 0.0, \left.\frac{\partial(w_2\hat{y}_2)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(n-a_k+1)}} = 0.0, \ldots, \left.\frac{\partial(w_k\hat{y}_k)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(n-a_k+1)}} = \frac{\partial y^{(n-a_k+1)}}{\partial x_d} \\
\quad \vdots \\
\left.\frac{\partial(w_1\hat{y}_1)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(n)}} = 0.0, \left.\frac{\partial(w_2\hat{y}_2)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(n)}} = 0.0, \ldots, \left.\frac{\partial(w_k\hat{y}_k)}{\partial x_d}\right|_{\mathbf{x}=\mathbf{x}^{(n)}} = \frac{\partial y^{(n)}}{\partial x_d} \qquad d = 1, 2, \ldots, m.
\end{cases}
\tag{24}
$$

Similarly, although there might be various type of ways for choosing qualified weight coefficient, we still follow the steps of build a special weight coefficient kriging to obtain w_i. Recall eq.(6) and eq.(7), a WCK for any untried $\mathbf{x}$ is built through the data set $\left(\mathbf{S}, \hat{\mathbf{y}}_{\mathrm{S,sub}}\right)$. Note that each submodel interpolates all n samples, which means the predictor of i-th submodel $\hat{y}_i$ is equal to the observed value $\mathbf{y}_{\mathrm{S}}$ at the sampling sites. That is to say, the WCK model is built through the data set $\left(\mathbf{S}, \mathbf{y}_{\mathrm{S}}\right)$ in fact and provides the weight coefficients both suitable for summing up all the function values or the predictors of submodels when building a WGEK. As the difference between the submodel of a WGEK and an

AWGEK is the number of sampling sites where local derivatives are considered, we could simply add the weight coefficients of these chosen sampling sites in one submodel up to ensure this submodel satisfies its own tangency condition. For clarity, $\bar{\mathbf{w}} = \left[\bar{w}_1, \bar{w}_2, ..., \bar{w}_n\right]^{\mathrm{T}}$ denotes the weight coefficients of a WCK model here, and can be found by solving the following linear equations:

$$\begin{bmatrix} \mathbf{R} & \mathbf{F} \\ \mathbf{F}^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \bar{\mathbf{w}} \\ \mu \end{bmatrix} = \begin{bmatrix} \mathbf{r} \\ 1 \end{bmatrix}, \tag{25}$$

where μ is the Lagrange multiplier, and $\mathbf{F} = \left[1, ..., 1\right]^{\mathrm{T}} \in \mathbb{R}^n$ for an ordinary kriging. Once the optimal $\bar{\mathbf{w}}$ is obtained, the weight coefficients for summing up the submodels of an AWGEK is given by

$$w_i = \sum_{p=1}^{a_i} \bar{w}_{(\sum_{j=1}^{i} a_j) - a_i + p}, \quad i = 1, 2, ..., k \ . \tag{26}$$

After determining the submodels and weight coefficients, the resulting AWGEK predictor is of the form

$$\hat{y}(\mathbf{x}) = \beta_0 + \mathbf{r}^{\mathrm{T}}(\mathbf{x}) \mathbf{R}^{-1} \left[\hat{\mathbf{y}}_{\mathbf{S},sub}(\mathbf{x}) - \beta_0 \mathbf{F} \right], \tag{27}$$

where

$$\begin{aligned} &\hat{\mathbf{y}}_{\mathbf{S},sub}(\mathbf{x}) = \left[\hat{y}_1(\mathbf{x}), \hat{y}_2(\mathbf{x}), ..., \hat{y}_k(\mathbf{x})\right]^{\mathrm{T}}, \\ &\beta_0 = \left(\mathbf{F}^{\mathrm{T}} \mathbf{R}^{-1} \mathbf{F}\right)^{-1} \mathbf{F}^{\mathrm{T}} \mathbf{R}^{-1} \hat{\mathbf{y}}_{\mathbf{S},sub}(\mathbf{x}). \end{aligned} \tag{28}$$

Similar to the WGEK, with the assumption that the prediction of the submodels can be generally considered to be normal distributed $\hat{Y}_i \sim \left[\hat{y}_i, s_i^2\right], i = 1, 2, ..., k$, the MSE of an AWGEK can be derived according to the theory of combining a group of normal distributions with different mean values and variances[49]:

$$MSE\left[\hat{y}(\mathbf{x})\right] = s^2(\mathbf{x}) = \left(\sum_{i=1}^{k} w_i s_i \right)^2, \tag{29}$$

where s_i can be obtained by eq.(22).

2.4 Correlation Functions and Hyperparameter Tuning

When using an AWGEK, it is necessary to set the correlation function and tune the hyperparameters to enhance the model accuracy and robustness. As is mentioned before, the correlation function determines the correlation and covariance between the responses of sample points[52], and here we focus on a family of correlation functions that are of the form

$$R(\boldsymbol{\theta}, \mathbf{x}, \mathbf{x}') = \prod_{i=1}^{m} scf\left(\theta_i, \left|x_i - x_i'\right|\right), \tag{30}$$

where "scf" denotes the spatial correlation function that only depends on the Euclidean distance between two sites ($\mathbf{x}$ and $\mathbf{x}'$) and the hyperparameters $\boldsymbol{\theta}$. Thus far, several types of correlation functions can be used by AWGEK, such as "Gaussian function", "cubic spline function", "Matérn function", etc. Note that, for the success of an AWGEK, the correlation function must be at least second order differentiable; otherwise, the use of gradients will not enhance the prediction[49]. As a matter of experience, the cubic spline function behaves well in the optimization problems and is employed in this paper:

$$scf_k \equiv scf\left(\theta_i, \left|x_i - x_i'\right|\right) \begin{cases} 1 - 15\xi_i^2 + 30\xi_i^3 & \text{for } 0 \le \xi_i \le 0.2 \\ 1.25\left(1 - \xi_i\right)^3 & \text{for } 0.2 < \xi_i < 1, \\ 0 & \text{for } \xi_i \ge 1 \end{cases} \tag{31}$$

where $\xi_i = \theta_i \left| x_i - x_i' \right|,\ i = 1,2,...,m$. The detailed derivation of the first-order and second-order partial derivative of cubic spline function can be referred to Ref.[49].

After the correlation function is defined, we will focus on the method of tuning the hyperparameters of AWGEK. Due to the reason that all the submodels pass through the n existing sample points and correspond to the same random process, we can reasonably assume that all the submodels share the same $\boldsymbol{\theta}, \sigma^2, \beta_0$. For the i-th submodel, $i \in \{1,2,...,k\}$, the likelihood function is

$$L_i(\beta_0, \sigma^2, \boldsymbol{\theta}) = \frac{1}{\sqrt{(2\pi\sigma^2)^{(n+a_i \cdot m)} \left| \overline{\mathbf{R}}_i \right|}} \exp\left(-\frac{1}{2} \frac{\left((\mathbf{y}_{\mathbf{S},i} - \beta_0 \overline{\mathbf{F}}_i)^{\mathrm{T}} \overline{\mathbf{R}}_i^{-1} (\mathbf{y}_{\mathbf{S},i} - \beta_0 \overline{\mathbf{F}}_i) \right)}{\sigma^2} \right), \tag{32}$$

Taking the logarithm, one can get

$$\ln L_i = -\frac{n + a_i \cdot m}{2} \ln(\sigma^2) - \frac{1}{2} \ln \left| \overline{\mathbf{R}}_i \right| - \frac{1}{2} \frac{\left(\mathbf{y}_{\mathbf{S},i} - \beta_0 \overline{\mathbf{F}}_i \right)^{\mathrm{T}} \overline{\mathbf{R}}_i^{-1} \left(\mathbf{y}_{\mathbf{S},i} - \beta_0 \overline{\mathbf{F}}_i \right)}{\sigma^2} - \frac{n + a_i \cdot m}{2} \ln(2\pi). \tag{33}$$

Then a uniform aggregation is used to get the joint logarithm likelihood function as

$$\begin{aligned} &Max. \qquad JL = \frac{1}{k} \left(\ln L_1 + \ln L_2 + \cdots + \ln L_k \right) \\ &w.r.t. \qquad \beta_0, \sigma^2, \boldsymbol{\theta} \end{aligned} \tag{34}$$

Taking the partial derivatives of JL with respect to β_0 and σ^2, and forcing them to zero, we obtain the following equations:

$$\begin{cases} \dfrac{\partial JL}{\partial \beta_0} = \dfrac{1}{k} \displaystyle\sum_{i=1}^{k} \frac{\overline{\mathbf{F}}_i^{\mathrm{T}} \overline{\mathbf{R}}_i^{-1} \left(\mathbf{y}_{\mathbf{S},i} - \beta_0 \overline{\mathbf{F}}_i \right)}{\sigma^2} = 0, \quad i = 1, \cdots, k \\ \dfrac{\partial JL}{\partial \sigma^2} = -\dfrac{1}{2k\sigma^2} \displaystyle\sum_{i=1}^{k} \left(n + a_i \cdot m \right) + \dfrac{1}{2k} \displaystyle\sum_{i=1}^{k} \frac{\left(\mathbf{y}_{\mathbf{S},i} - \beta_0 \overline{\mathbf{F}}_i \right)^{T} \overline{\mathbf{R}}_i^{-1} \left(\mathbf{y}_{\mathbf{S},i} - \beta_0 \overline{\mathbf{F}}_i \right)}{(\sigma^2)^2} = 0. \end{cases} \tag{35}$$

Then, the analytical optimum of β_0 and σ^2 can be given by

$$\begin{aligned} \beta_0(\boldsymbol{\theta}) &= \sum_{i=1}^{k} \left(\overline{\mathbf{F}}_i^{\mathrm{T}} \overline{\mathbf{R}}_i^{-1} \mathbf{y}_{\mathbf{S},i} \right) \Big/ \sum_{i=1}^{k} \left(\overline{\mathbf{F}}_i^{\mathrm{T}} \overline{\mathbf{R}}_i^{-1} \overline{\mathbf{F}}_i \right), \\ \sigma^2(\beta_0, \boldsymbol{\theta}) &= \frac{1}{k(n + a_i \cdot m)} \sum_{i=1}^{k} \left(\mathbf{y}_{\mathbf{S},i} - \beta_0 \overline{\mathbf{F}}_i \right)^{\mathrm{T}} \overline{\mathbf{R}}_i^{-1} \left(\mathbf{y}_{\mathbf{S},i} - \beta_0 \overline{\mathbf{F}}_i \right). \end{aligned} \tag{36}$$

As there is no closed form solution for $\boldsymbol{\theta}$, it must be found by maximizing the following concentrated joint logarithm likelihood function using numerical optimization

$$\boldsymbol{\theta} = \arg \max_{\boldsymbol{\theta} > 0} \left[-\sum_{i=1}^{k} (n + a_i \cdot m) \ln(\sigma^2) - \sum_{i=1}^{n} \ln \left| \overline{\mathbf{R}}_i \right| \right]. \tag{37}$$

In this article, an improved version of Hooke &Jeeves pattern search method (by using multi-starts search and a trust-region method) is used to solve the preceding optimization problem. Moreover, we normalize the input variables $\mathbf{x}$ to the range [0.0, 1.0] and limit the searching interval of optimal $\boldsymbol{\theta}$ to the range [0.005, 0.618] for the cubic spline function, according to our intensive numerical experiments, to make the model more robust.

2.5 Discussion about the Benefit of Proposed Method

Compared with GEK and WGEK, which introduces gradients at all sampling sites, and gradients at only one sampling site in each submodel respectively, AWGEK is more like in an intermediate state. It determines the number of required submodels adaptively according to the dimension of the problem and the number of current sample points. Each submodel introduces gradients at more than one sampling site when there exist numerous sample points $(n \gg m)$. Otherwise, the AWGEK will

degenerate into the original WGEK if the number of sample points n is close to the dimension m. Assuming that the computation cost of optimizing hyperparameters could be ignored, the efficiency of building a GEK, WGEK or AWGEK model is then compared by the FLOPs of decomposing the correlation matrices, which occupies a sizeable proportion in the cost of training a surrogate model. Recall that the FLOPs of decomposing all the correlation matrices of a GEK and WGEK model by Cholesky method is around $O(n^3(1+m)^3)$ and $O(n(n+m)^3)$ respectively. In contrast, when training an AWGEK, we need to decompose k much smaller correlation matrices, whose size is as small as $(n+a_i m)\times(n+a_i m)$ for the i-th submodel, and the total FLOPs is around $O(\sum_{i=1}^{k}(n+a_i m)^3)$. The comparison of FLOPs with different numbers of dimensions and sampling sites for building an AWGEK and WGEK model is plotted in Figure 1 and Table 1. It can be seen that the cost of training both a WGEK or an AWGEK raises quickly with the increase of numbers of dimensions and sampling points. However, the growth rate of the training cost for AWGEK is slower than that of WGEK under the same dimension especially when there exist numerous sample points. This is benefit to an expensive high-dimensional aerodynamic modeling or optimization problem, for the dimension of the problem is fixed during the optimization process and the number of samples could become quite large in the terminal stage.

In addition, the accuracy of WGEK is somehow decreased compared to a GEK, as a price paid for the large efficiency improvement[49]. While for an AWGEK, the correlation matrix reserves second-order cross partial derivative between several sample points inside each submodel, as is formulated in eq.(20). As a result, the model accuracy could be slightly enhanced compared to a WGEK. This will be verified by some numerical examples in the following section.

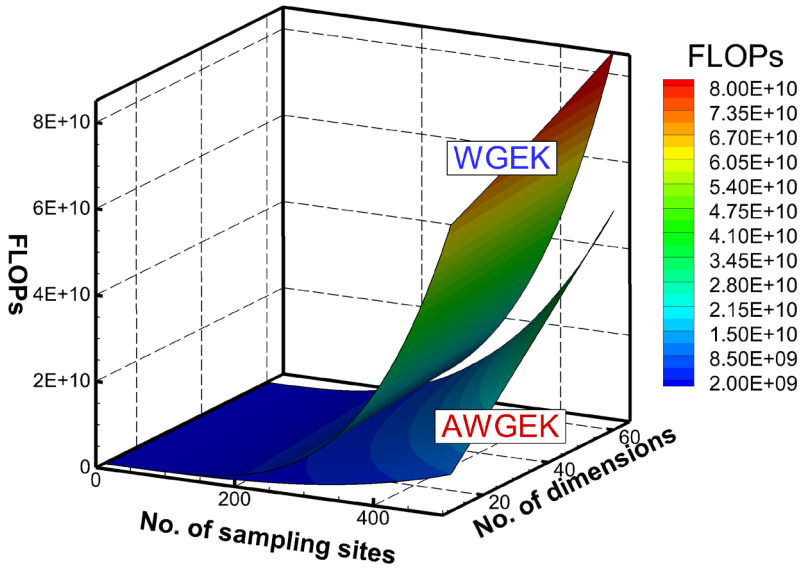


Figure 1 Comparison of FLOPs of decomposing the correlation matrices of WGEK and proposed AWGEK model

Table 1 Speed-up ratios of proposed AWGEK against WGEK for typical numbers of dimensions and samples

Number of dimensions (m)	Number of sample points (n)	Theoretical speed-up ratio $n(n+m)^3 \Big/ \sum_{i=1}^{k}(n+a_i m)^3$
5	10	1.000
5	20	1.153
10	50	1.262
20	120	1.408
50	350	1.537

50	600	2.257
50	1000	3.423
100	3000	4.898

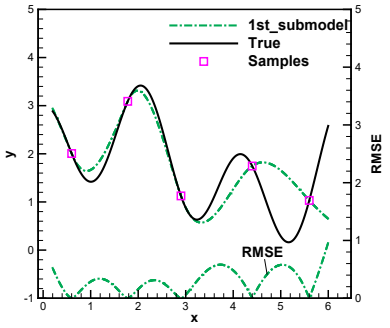
3. Numerical Examples Validation

3.1 One-Dimensional Analytical Test Case

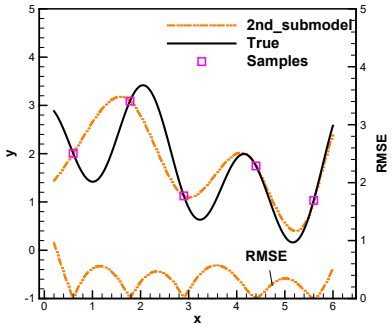
A one-dimensional analytical function from Ref.[49] is taken as the first test case to illustrate the procedure of building an AWGEK and to verify its correctness:

$$f(x) = e^{-x} + \sin(x) + \cos(3x) + 0.2x + 1.0, \quad x \in [0.2, 6.0] \tag{38}$$

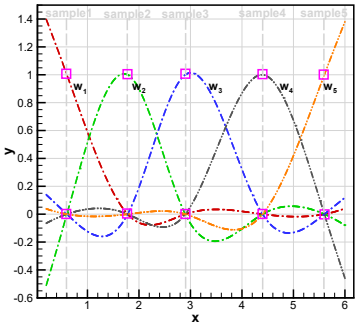
Five sampling sites, marked with pink squares $x = \{0.60, 1.78, 2.90, 4.40, 5.60\}$, are chosen along with gradients computed analytically. The procedure of building an AWGEK is sketched in Figure 2. The first submodel is built using the functional values at all the sampling sites and three gradients at the first three sampling sites, while the second submodel is built using the functional values at all the sampling sites and two gradients at the last two sampling sites. Note that the grouping of the gradients at five sampling sites, which introduces three and two gradients in each model respectively, is totally adaptive according to the grouping mechanism in section 2.2 for the purpose of reaching the smallest total FLOPs. Certainly, to choose which three sampling sites from all five samples could also be a problem and here we just choose from the data set by sequence. This might affect the final accuracy of AWGEK but is beyond the scope of this article. One can see that each submodel passes through all the sample points and features the observed gradients at its own grouped sampling sites. Once the two submodels are built, five weight coefficient curves are defined by the WCK model, as is sketched in Figure 2(c). Afterwards, we add the first three weight coefficients to obtain the appropriate weight coefficient for summing up the first submodel and add the last two weight coefficients to obtain the appropriate weight coefficient for summing up the second submodel respectively, as is plotted in Figure 2(d). Both interpolation and tangency conditions could be satisfied through this weighting method. The final predicted curve by AWGEK is sketched in Figure 2(e) and it precisely features the observed functional values and gradients at all the sample sites, which verifies the correctness of the proposed method. The predicted curve along with root mean square error (RMSE) by WGEK are also plotted in Figure 2(e), which shows that the AWGEK model is comparable to the WGEK model, and even more accurate in the valleys of the true function.



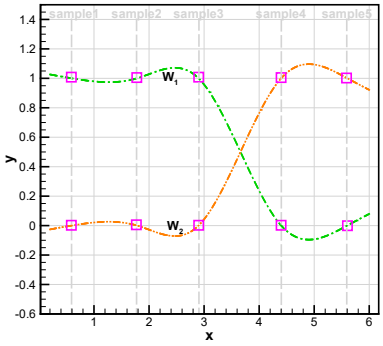
(a) First submodel (five function values + three gradients at first three sampling sites)



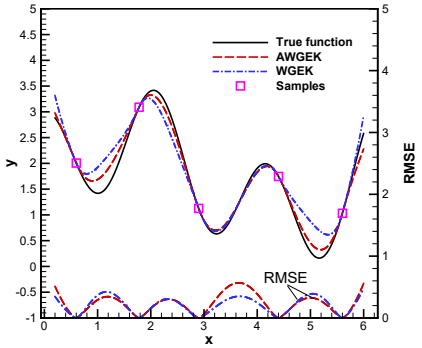
(b) Second submodel (five function values + two gradients at last two sampling sites)



(c) Weight coefficients obtained by WCK model



(d) Weight coefficients for summing up two submodels (interpolation and tangency conditions satisfied)



(e) Resulting AWGEK model and comparison with WGEK model ($\theta_{AWGEK} = \theta_{WGEK} = 0.174$)

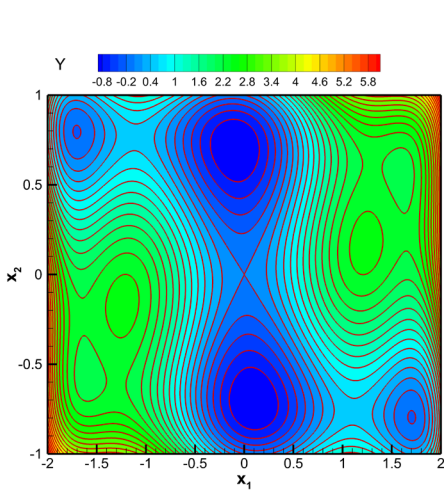
Figure 2 Illustration of building an AWGEK for a one-dimensional test function

3.2 Two-Dimensional Analytical Test Case

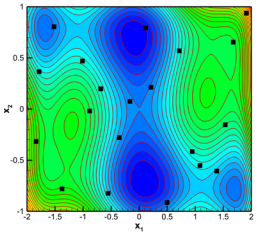
Then, a two-dimensional six-hump camelback function is employed to validate our proposed method.

$$f(\mathbf{x}) = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + \left(-4 + 4x_2^2\right)x_2^2, \qquad (39)$$
$$x_1 \in [-2,2], \quad x_2 \in [-1,1]$$

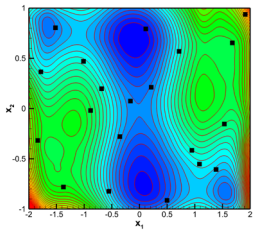
20 sampling sites are selected by Latin Hypercubic Sampling (LHS) method for AWGEK model modeling, and the final prediction results are plotted in Figure 3. As shown in Figure 3(c), the response surface predicted by AWGEK indicates the regions where two global and five local optimal solutions locate. Although its prediction accuracy is slightly lower than that of the GEK in Figure 3(b), it is improved compared with that of the WGEK model in Figure 3(d), which could only find the regions where two global and four local optimal solutions are. Figure 3 also shows the comparison of predicted RMSEs of the three gradients-enhanced models. We can see that the RMSEs of the AWGEK is larger than that of the GEK, but slightly smaller than that of the WGEK model, which indicates that the model accuracy of AWGEK is between that of the GEK and WGEK model.



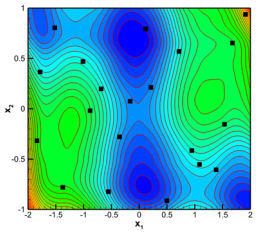
(a) True function



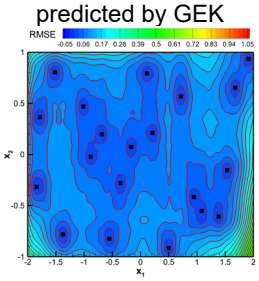
(b) Response surface predicted by GEK



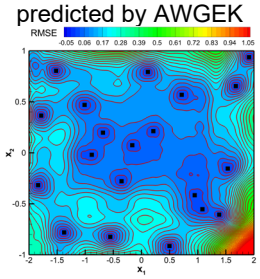
(c) Response surface predicted by AWGEK



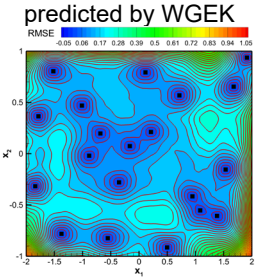
(d) Response surface predicted by WGEK



(e) RMSE predicted by GEK



(f) RMSE predicted by AWGEK



(g) RMSE predicted by WGEK

Figure 3 Comparison of response surfaces and RMSEs by GEK, WGEK and AWGEK models for a two-dimensional six-hump camelback test function

3.3 High-Dimensional Analytical Test Case

3.3.1 Varied-Dimensional Sum-Square Test Case

In this subsection, the following sum-square function is adopted to evaluate the accuracy loss of the

proposed AWGEK model against GEK and WGEK for higher-dimensional problems with respect to the variation of sampling sites:

$$f(\mathbf{x}) = \sum_{i=1}^{m} i x_i^2, \quad x \in [-5,5], \quad i = 1,\ldots,m \tag{40}$$

where m is the number of dimensions and $m = 5,10,20$ in this subsection. A correlation coefficient r^2 is introduced to assess the accuracy of the kriging, GEK, WGEK and AWGEK models:

$$r^2 = \left(\frac{\sigma_{f\hat{f}}}{\sqrt{\sigma_f \sigma_{\hat{f}}}} \right)^2 = \left(\frac{N\sum f\hat{f} - \sum f \sum \hat{f}}{\sqrt{\left[N\sum f^2 - \left(\sum f\right)^2 \right]\left[N\sum \hat{f}^2 - \left(\sum \hat{f}\right)^2 \right]}} \right)^2, \tag{41}$$

and the surrogate model is perfectly accurate when r^2=1.0, whereas r^2=0.0 indicates a bad approximation. All the model fitting experiments of numerical examples in this section are performed with a PC-based workstation (Intel Xeon CPU E5-1620, 8-cores, 3.5 GHz).

For 5, 10, and 20-dimensional sum-square functions, the surrogate models are built using increasing numbers of sample sites, and the growth of the correlation coefficients are plotted in Figure 4. It is shown that the conventional GEK is the most accurate model among all these examined models and the AWGEK is slightly more accurate than the WGEK model. This is not a surprise because the correlation matrix of the GEK model is the most complete, containing all the information about spatial correlations of different sampling sites and their gradients, while the AWGEK model removes the spatial correlations between the gradients of different submodels at sampling sites, which results in a decrease in the model accuracy. However, within each submodel of the AWGEK, the spatial correlations between the gradients at the sampling points is retained, for it guarantees at least some cross-terms of second derivatives in the submodels. Hence, the AWGEK is still accurate than the WGEK model. Note that all the gradient-enhanced models are much more accurate than the original kriging. Table 2 shows the comparison of training cost between the WGEK and AWGEK for modeling varied-dimensional sum-square test functions. It can be seen that, along with the increase of numbers of dimensions and sampling points, the cost of training a WGEK raises quickly while the cost of training an AWGEK raises slowly and is much lower especially when existing numerous sample points.

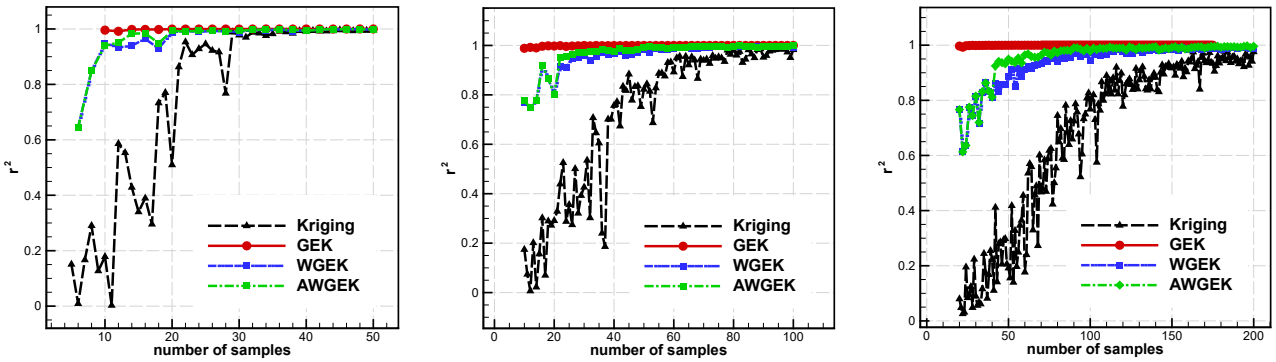


Figure 4 Evaluation of kriging, GEK, WGEK and AWGEK models for varied-dimensional Sum-Square test functions (left: 5-dimensional; middle: 10-dimensional; right: 20-dimensional)

Table 2 Comparison of WGEK and AWGEK for varied-dimensional Sum-Square test case

Number of dimensions (m)	Number of sample points (n)	Number of calculating MLE		Model fitting time/sec.		
		WGEK	AWGEK	WGEK	AWGEK	Speed-up ratio
20	60	3970	2993	199.960	183.765	1.088
20	100	5896	3437	783.999	471.944	1.661
30	150	11590	10899	8033.731	6576.972	1.221
50	200	14561	13455	37763.99	13439.19	2.810

3.3.2 Other Benchmark Numerical Test Cases

Two more benchmark numerical test cases, Dixon-Price function and Rosenbrock function[54], are employed in this subsection to further demonstrate the efficiency of our AWGEK modeling method.

$$\begin{aligned}&\text{Dixon-Price}\quad f(x)=(x_1-1)^2+\sum_{i=2}^{30} i(2x_i^2-x_{i-1})^2 \quad x_i\in[-10,10], i=1,2,...,30\\&\text{Rosenbrock}\quad f(x)=\sum_{i=1}^{49}\left[100(x_{i+1}-x_i^2)^2+(x_i-1)^2\right] \quad x_i\in[-5,10], i=1,2,...,50\end{aligned} \tag{42}$$

Apart from the correlation coefficient r^2 mentioned above, two other accuracy metrics, relative root-mean-square error (RRMSE) and relative maximum absolute error (RMAE), are calculated.

$$\begin{aligned}RRMSE &= \frac{\sqrt{\sum_{i=1}^{N}\left(f-\hat{f}\right)^2 \Big/ N}}{\sqrt{\sum_{i=1}^{N}\left(f-\bar{f}\right)^2 \Big/ (N-1)}},\\RMAE &= \frac{\max\left(\left|f-\hat{f}\right|\right)}{\sqrt{\sum_{i=1}^{N}\left(f-\bar{f}\right)^2 \Big/ (N-1)}}.\end{aligned} \tag{43}$$

Here, 1000 validation samples are selected by LHS, hence $N = 1000$ in eq.(43). Note that r^2 and RRMSE can reflect the global accuracy of the surrogates, while RMAE is a criterion that can indicate the local predicting performance of the surrogates[54]. The closer the value of r^2 is to 1, and the smaller values of RRMSE and RMAE, the more accurate the model is. To analyze the robustness of our method, the experiments are repeated 10 times with initial samples randomly chosen by LHS for each test case. Then the mean and standard deviation values of modeling time and three accuracy metrics are calculated.

Table 3 gives the comparison of model accuracy and modeling efficiency of WGEK and AWGEK for two benchmark numerical test cases. It can be seen that our proposed AWGEK model is more accurate than WGEK for all three metrics are better and the modeling efficiency is greatly improved.

Table 3 Comparison of WGEK and AWGEK for benchmark numerical test cases (repeat 10 times)

Function	Surrogate	Statistic	r^2	RRMSE	RMAE	Model fitting time/sec.
Dixon-Price (30D) with 300 Initial Samples	AWGEK	mean	**0.910843**	**0.565614**	**1.596069**	**5874.00**
		std	0.004707	0.020572	0.199816	1499.48
	WGEK	mean	0.909770	0.899179	2.218587	7065.32
		std	0.003960	0.018855	0.125519	845.720
Rosenbrock (50D) with 600 Initial Samples	AWGEK	mean	**0.924246**	**0.385997**	**1.187887**	**83400.918**
		std	0.005578	0.016690	0.073927	14955.211
	WGEK	mean	0.904526	0.631272	1.571794	110705.048
		std	0.004169	0.026058	0.027518	15916.108

4. Application to High-Dimensional Aerodynamic Modeling Problem

4.1 Problem Statement

In this part, the proposed method is validated through an engineering application. Here, we focus on an aerodynamic modeling problem of the RAE2822 airfoil in transonic regime with 36 design variables parameterized by free form deformation (FFD) method, which is sketched in Figure 5. Our objective is to train different gradient-based kriging models using transonic database (at a freestream Mach number of 0.734 and angle of attack of 2.738 degree, and Reynolds number of 6.5×10^6) to predict

the aerodynamic force and moment coefficients: C_l , C_d, and C_m.

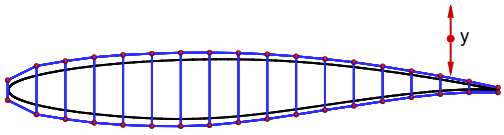


Figure 5 Shape design variables determined by the y displacements of 36 FFD control points

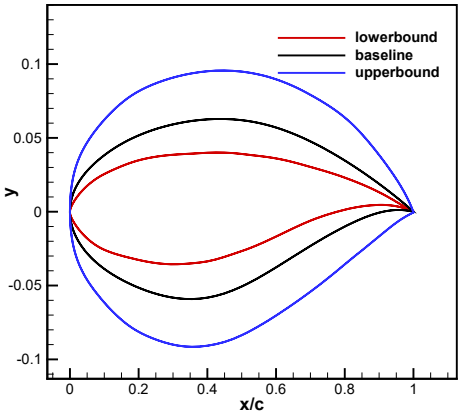
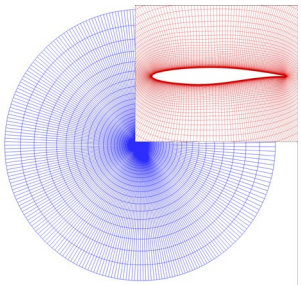


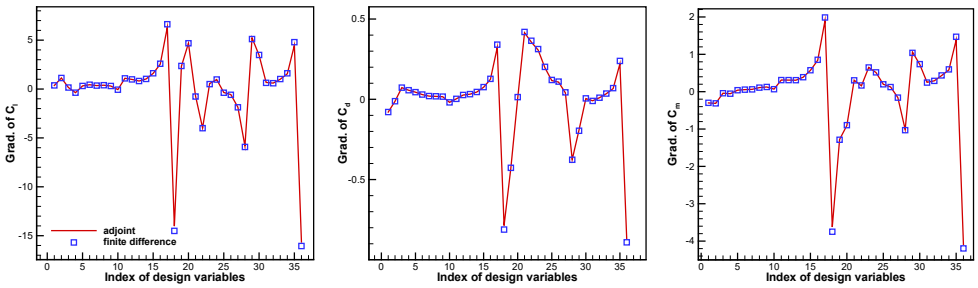
Figure 6 The design space and baseline airfoil for RAE2822 airfoil aerodynamic modeling case

4.2 Database Preparation

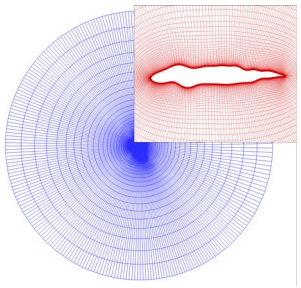
Figure 6 shows the design space for the RAE2822 airfoil aerodynamic modeling case and 400 sampling sites are selected by LHS method. The deformed airfoil aerodynamic force coefficients and their respective gradients are then computed using ADflow[55][56], which solves the RANS equations with a Spalart-Allmaras turbulence model. The CFD grids for all airfoils are automatically generated by the open-source package pyHyp with the grid distribution of 256(chordwisedirection) ×128(normaldirection) sketched in Figure 7. The flow solution residuals are converged by 12 orders of magnitude below the initial residual. Figure 7 also shows that computed gradients by the adjoint solver are in reasonably agreement with that by the finite difference method, for both baseline airfoil and randomly deformed airfoil respectively.



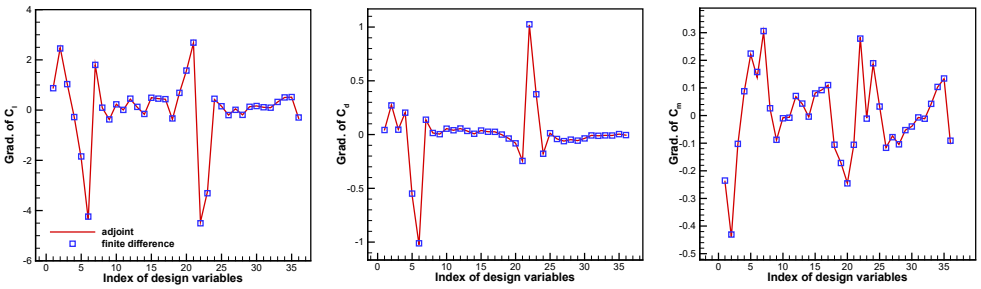
a) baseline airfoil



b) comparison of gradients obtained by adjoint and that by finite difference for baseline airfoil



c) randomly deformed airfoil



d) comparison of gradients obtained by adjoint and that by finite difference for a randomly deformed airfoil

Figure 7 The grids used for CFD simulation and the validation of adjoint solver by comparison of computed gradients and that obtained by finite difference for baseline airfoil and a randomly deformed airfoil respectively

The final number of sample points in training database is set to 549, getting rid of 51 samples with unconverged CFD solutions due to the extremely deformed and abnormal airfoil surfaces. The prediction accuracy of the gradient-enhanced models is evaluated with the correlation coefficient r^2 (mentioned in section 3.3.1), RRMSE and RMAE (mentioned in section 3.3.2) at 800 testing samples

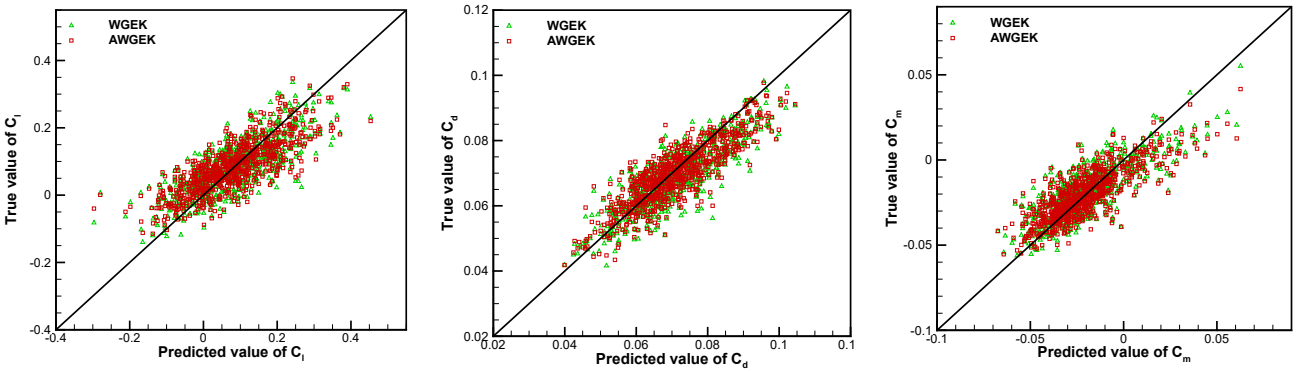
selected by Uniform Design (UD). All the model fitting experiments are performed with a PC-based workstation (Intel Xeon CPU E5-1620, 8-cores, 3.5 GHz).

4.3 Result Discussion

The comparison of our AWGEK with WGEK model in both model accuracy and fitting efficiency is given in Table 4. It is observed that, for lift coefficient and drag coefficient, the r^2 of the AWGEK are both larger than that of the WGEK and the RRMSE and RMAE also show that the AWGEK is slightly accurate than the WGEK model. However, for pitch-moment coefficient, the AWGEK does not perform well for all measurement indictors are slightly worse than that of the WGEK, which can be also seen in Figure 8. Figure 9 also shows the comparison of absolute error calculated for aerodynamic force coefficients with the mean error given by the red triangles. It can be seen that, the mean absolute error for each coefficient is close to each other, and the predicted results of the AWGEK is slightly better than that of the WGEK for C_l, C_d but worse for C_m. Note that for all three aerodynamic coefficients modeling and prediction, the model fitting time of the AWGEK model is much lower than that of the WGEK model, which demonstrates the fitting efficiency of our proposed AWGEK model in high-dimensional problem.

Table 4 Accuracy and efficiency measures for AWGEK and WGEK model in 36-dimensional RAE2822 airfoil aerodynamic modeling case

	Model	r^2	RRMSE	RMAE	Model fitting time/sec.
C_l	AWGEK	**0.741622**	**0.507962**	**2.086780**	**12166.646**
	WGEK	0.735437	0.514006	2.128090	25455.432
C_d	AWGEK	**0.992852**	**0.084487**	**0.298006**	**20914.836**
	WGEK	0.992169	0.088431	0.352148	34164.693
C_m	AWGEK	0.831988	0.409613	1.670950	**14178.830**
	WGEK	**0.837309**	**0.403075**	**1.460690**	16974.751



a) lift coefficient b) drag coefficient c) pitch-moment coefficient

Figure 8 Linear regressions of true relative to predicted aerodynamic coefficients

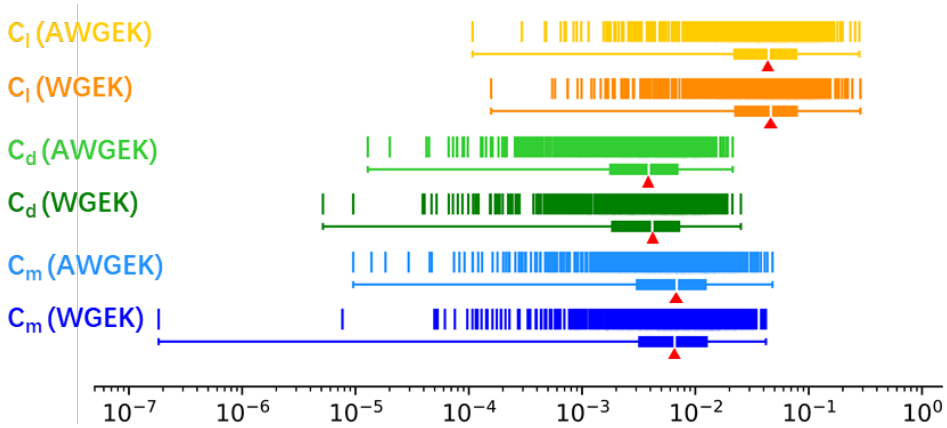


Figure 9 The comparison of absolute error calculated by AWGEK and WGEK respectively for C_l, C_d and C_m with the mean error given by the red triangles

5. Conclusion

In this paper, an improved formulation of weighted gradient enhanced kriging, called adaptive weighted gradient enhanced kriging (AWGEK), is proposed to further improve both the model accuracy and fitting efficiency of a gradient-enhanced surrogate especially when existing numerous training samples in a high-dimensional problem. It provides an adaptive grouping mechanism according to the dimension of the problem and the current number of training samples to interpolate gradients at more than one sampling stie in each submodel. Taking the total cost of correlation matrix decomposition into account, the main drawback of a WGEK that there would exist numerous need-to-build submodels corresponding to each sample can be avoided in the model fitting process.

Several numerical examples are employed to validate the correctness and effectiveness of AWGEK and the results show that the prediction accuracy of an AWGEK is better than that of a WGEK model, and it has obvious advantage in model fitting efficiency when existing numerous samples in training database. At last, the proposed AWGEK is applied to the aerodynamic modeling of an RAE2822 airfoil parameterized with 36 deign variables in transonic regime. It is observed that, the prediction of the AWGEK model for the lift and drag coefficients is more accurate than that of the WGEK while it is slightly worse for the pitch-moment coefficient. However, the cost for model training of the AWGEK model is much lower than that of the WGEK model, which shows that the proposed AWGEK has great potential for applications to higher-dimensional engineering design problems. In the future, we will apply the AWGEK model to design optimization problem, and focus on the performance study.

6. Acknowledgement

This research was supported by the National Natural Science Foundation of China grant number 11972305, the National Numerical Wind Tunnel Project Number 2019ZT6-A12 and 2018-ZT1A03, and was sponsored by Innovation Foundation for Doctor Dissertation of Northwestern Polytechnical University CX202017. The authors also would like to thank Hao-Dong Dang, Ming-Qi Liu for their valuable help and useful discussions.

7. Contact Author Email Address

Zhong-Hua Han*, Professor, hanzh@nwpu.edu.cn, corresponding author.

8. Copyright Statement

References

[1] Simpson T W, Peplinski J, Koch P N, et al. Metamodels for computer-based engineering design: survey and recommendations. *Engineering with Computers*, Vol. 17, No. 2, pp 129-150, 2001.

[2] Queipo N V, Haftka R T, Shyy W, et al. Surrogate-based analysis and optimization. *Progress in Aerospace Sciences*, Vol. 41, No. 1, pp 1-28, 2005.

[3] Forrester A I J, Keane A J. Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences*, Vol. 45, No. 1, pp 50-79, 2009.

[4] Viana F A C, Simpson T W, Balabanov V, et al. Metamodeling in multidisciplinary design optimization: how far have we really come? *AIAA Journal*, Vol. 52, No. 4, pp 670-690, 2014.

[5] Li J C, Cai J S, Qu K. Surrogate-based aerodynamic shape optimization with the active subspace method. *Structural and Multidisciplinary Optimization*, Vol. 59, No. 2, pp 403-416, 2019.

[6] Shi R H, Liu L, Long T, et al. Filter-based adaptive Kriging method for black-box optimization problems with expensive objective and constraints. *Computer Methods in Applied Mechanics and Engineering*, Vol. 347, pp 782-805, 2019.

[7] Shi R H, Liu L, Long T, et al. Multidisciplinary modeling and surrogate assisted optimization for satellite constellation systems. *Structural and Multidisciplinary Optimization*, Vol. 58, No. 5, pp 2173-2188, 2018.

[8] Bu Y P, Song W P, Han Z H. Aerodynamic/aeroacoustic variable-fidelity optimization of helicopter rotor based on hierarchical Kriging model. *Chinese Journal of Aeronautics*, Vol. 33, No. 2, pp 476-492, 2020.

[9] Liu F, Han Z H, Zhang Y, et al. Surrogate-based aerodynamic shape optimization of hypersonic flows considering transonic performance. *Aerospace Science and Technology*, Vol. 93, pp 105345, 2019.

[10] Han S Q, Song W P, Han Z H, et al. Hybrid inverse/optimization design method for rigid coaxial rotor airfoils considering reverse flow. *Aerospace Science and Technology*, Vol. 95, pp 105488, 2019.

[11] Han Z H, Chen J, Zhang K S, et al. Aerodynamic shape optimization of natural-laminar-flow wing using surrogate-based approach. *AIAA Journal*, Vol. 56, No. 7, pp 2579-2593, 2018.

[12] Jones D R, Schonlau M, Welch W J. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, Vol. 13, No. 4, pp 455-492, 1998.

[13] Chernukhin O, Zingg D W. Multimodality and global optimization in aerodynamic design. *AIAA Journal*, Vol. 51, No. 6, pp 1342-1354, 2013.

[14] Liu J, Song W P, Han Z H, et al. Efficient aerodynamic shape optimization of transonic wings using a parallel infilling strategy and surrogate models. *Structural and Multidisciplinary Optimization*, Vol. 55, No. 3, pp 925-943, 2017.

[15] Han Z H. SurroOpt: A generic surrogate-based optimization code for aerodynamic and multidisciplinary design. *30th International Council of the Aeronautical Sciences*, Daejeon, Korea, ICAS-2016-0281, pp 1-10, 2016.

[16] Vavalle A, Qin N. Iterative response surface based optimization scheme for transonic airfoil design. *Journal of Aircraft*, Vol. 42, No. 2, pp 413-420, 2005.

[17] Sobester A, Leary S J, Keane A J. On the design of optimization strategies based on global response surface approximation models. *Journal of Global Optimization*, Vol. 33, No. 1, pp 31-59, 2005.

[18] Krige D G. A statistical approach to some basic mine valuation problems on the Witwatersrand. *Journal of the Southern African Institute of Mining and Metallurgy*, Vol. 52, No. 6, pp 119-139, 1951.

[19] Sacks J, Welch W J, Mitchell T J, et al. Design and analysis of computer experiments. *Statistical science*, Vol. 4, No. 4, pp 409-435, 1989.

[20] Tao J, Sun G, Guo L Q, et al. Application of a PCA-DBN-based surrogate model to robust aerodynamic design optimization. *Chinese Journal of Aeronautics*, Vol. 33, No. 6, pp 1573-1588, 2020.

[21] Zhang X S, Xie F F, Ji T W, et al. Multi-fidelity deep neural network surrogate model for aerodynamic shape optimization. *Computer Methods in Applied Mechanics and Engineering*, Vol. 373, pp 113485, 2021.

[22] Yun Y, Yoon M, and Nakayama H. Multi-objective optimization based on meta-modeling by using support vector regression. *Optimization and Engineering*, Vol. 10, No. 2, pp 167-181, 2009.

[23]Wang Q, Moin P, Iaccarino G. A high-order multi-variate approximation scheme for arbitrary data sets. *Journal of Computational Physics*, Vol. 229, No. 18, pp 6343-6361, 2010.

[24]Crestaux T, Maître O L, Martinez J M. Polynomial chaos expansion for sensitivity analysis. *Reliability Engineering and System Safety*, Vol. 94, No. 7, pp 1161-1172, 2009.

[25]Barber D. *Bayesian reasoning and machine learning*. Cambridge University Press, 2012.

[26]Song W B, Keane A J. Surrogate-based aerodynamic shape optimization of a civil aircraft engine nacelle. *AIAA Journal*, Vol. 45, No. 10, pp 2565-2574, 2007.

[27]Li J C, Cai J S. Massively multipoint aerodynamic shape design via surrogate-assisted gradient-based optimization. *AIAA Journal*, Vol. 58, No. 5, pp 1949-1963, 2020.

[28]Zhang Y, Han Z H, Leifsson L T. Surrogate-based optimization applied to benchmark aerodynamic design problems. *35th AIAA Applied Aerodynamics Conference*, Denver, Colorado, AIAA-2017-4367, pp 1-15, 2017.

[29]Raponi E, Bujny M, Olhofer M, et al. Kriging-assisted topology optimization of crash structures. *Computer Methods in Applied Mechanics and Engineering*, Vol. 348, pp 730-752, 2019.

[30]Xu C Z, Han Z H, Zhang K S, et al. Surrogate-based optimization method applied to multidisciplinary design optimization architectures. *31st International Council of the Aeronautical Sciences*, Belo Horizonte, Brazil, ICAS-2018-0496, pp 1-10, 2018.

[31]Shan S, Wang G G. Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions. *Structural and Multidisciplinary Optimization*, Vol. 41, No. 2, pp 219-241, 2010.

[32]Koch P N, Simpson T W, Allen J K, et al. Statistical approximations for multidisciplinary design optimization: the problem of the size. *Journal of Aircraft*, Vol. 36, No. 1, pp 275-286, 1999.

[33]Rabitz H, Alis O F. General foundations of high-dimensional model representations. *Journal of Mathematical Chemistry*, Vol. 25, pp 197-223.

[34]Ulaganathan S, Couckuyt I, Dhaene T, et al. High dimensional kriging metamodelling utilising gradient information. *Applied Mathematical Modelling*, Vol. 40, pp 5256-5270, 2016.

[35]Chen L M, Qiu H B, Gao L, et al. A screening-based gradient-enhanced kriging modeling method for high-dimensional problems. *Applied Mathematical Modelling*, Vol. 69, pp 15-31, 2018.

[36]Fernandez G, Park C Y, Kim N H, et al. Issues in deciding whether to use multifidelity surrogates. *AIAA Journal*, Vol. 57, No. 5, pp 2039-2054, 2019.

[37]Park C Y, Haftka R T, Kim N H. Remarks on multi-fidelity surrogates. *Structural and Multidisciplinary Optimization*, Vol. 55, No. 3, pp 1029-1050, 2017.

[38]Han Z H, Zimmermann R, Görtz S. Alternative cokriging model for variable-fidelity surrogate modeling. *AIAA Journal*, Vol. 50, No. 5, pp 1205-1210, 2012.

[39]Han Z H, Xu C Z, Zhang L, et al. Efficient aerodynamic shape optimization using variable-fidelity surrogate models and multilevel computational grids. *Chinese Journal of Aeronautics*, Vol. 33, No. 1, pp 31-47, 2020.

[40]Chung H S, Alonso J J. Using gradients to construct cokriging approximation models for high-dimensional design optimization problems. *40th AIAA Aerospace Sciences Meeting & Exhibit*, Reno, NV, AIAA-2002-0317, pp 1-15, 2002.

[41]Han Z H, Görtz S, Zimmermann R. Improving variable-fidelity surrogate modeling via gradient-enhanced kriging and a generalized hybrid bridge function. *Aerospace Science and Technology*, Vol. 25, No. 1, pp 177-189, 2013.

[42]Jameson A. Aerodynamic design via control theory. *Journal of Scientific Computing*, Vol. 3, No. 3, pp 233-260, 1988.

[43]Morris M D, Mitchell T J, Ylvisaker D. Bayesian design and analysis of computer experiments: use of derivatives in surface prediction. *Technometrics*, Vol. 35, No. 3, pp 243-255, 1993.

[44]Laurent L, Le Riche R, Soulier B, et al. An overview of gradient-enhanced metamodels with applications. *Archives of Computational Methods in Engineering*, Vol. 26, pp 61-106, 2019.

[45]Liu W Y, Batill S. Gradient-enhanced response surface approximations using kriging models. *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Atlanta, Georgia, AIAA-2002-5456, pp 1-11, 2002.

[46]Laurenceau J, Sagaut P. Building efficient response surfaces of aerodynamic functions with kriging and cokriging. *AIAA Journal*, Vol. 46, No. 2, pp 498-507, 2008.

[47]Bouhlel M A, Martins J R R A. Gradient-enhanced kriging for high-dimensional problems. *Engineering with Computers*, Vol. 35, pp 157-173, 2019.

[48]Song C, Song W P, Yang X D. Gradient-enhanced hierarchical kriging model for aerodynamic design optimization. *Journal of Aerospace Engineering*, Vol. 30, No. 6, pp 04017072, 2017.

[49]Han Z H, Zhang Y, Song C X, et al. Weighted gradient-enhanced kriging for high-dimensional surrogate modeling and design optimization. *AIAA Journal*, Vol. 55, No. 12, pp 4330-4346, 2017.

[50]Laurenceau J, Meaux M, Montagnac M, et al. Comparison of gradient-based and gradient-enhanced response-surface-based optimizers. *AIAA Journal*, Vol. 48, No. 5, pp 981-994, 2010.

[51]Zimmermann R. On the maximum likelihood training of gradient-enhanced spatial gaussian processes. *Siam Journal on Scientific Computing*, Vol. 35, No. 6, pp 2554-2574, 2013.

[52]Palar P S, Zuhal L R, Shimoyama K. Gaussian process surrogate model with composite kernel learning for engineering design. *AIAA Journal*, Vol. 58, No. 4, pp 1864-1880, 2020.

[53]Liu J, Han Z H, Song W P. Comparison of infill sampling criteria in Kriging-based aerodynamic optimization. *28th International Council of the Aeronautical Sciences*, Brisbane, Australia, ICAS 2012, pp 1625-1634, 2012.

[54]Zhao L, Wang P, Song B W, et al. An efficient kriging modeling method for high-dimensional design problems based on maximal information coefficient. *Structural and Multidisciplinary Optimization*, Vol. 61, pp 39-57, 2020.

[55]Yildirim A, Kenway G K W, Mader C A, et al. A Jacobian-free approximate Newton–Krylov startup strategy for RANS simulations. *Journal of Computational Physics*, Vol. 397, pp 108741, 2019.

[56]Mader C A, Kenway G K W, Yildirim A, et al. ADflow: An open-source computational fluid dynamics solver for aerodynamic and multidisciolinary optimization. *Journal of Aerospace Information Systems*, Vol.17, No. 9, pp 508-527, 2020.