Copyright © 2014 Tech Science Press CMES, vol.103, no.6, pp.429-462, 2014

Global Approximation for a Simulation Model Based on the RBF Response Surface Set

Yin Xiao-Liang[1], Wu Yi-Zhong[1,2], Wan Li[1] and Xiong Hui-Yuan[3]

Abstract: The use of multi-dimensional global approximation for a complex black-box function (such as a simulation or an analysis model) is steadily growing in the past decade. It can be applied in many fields such as parameter experiment, sensibility analyses real-time simulation, and design/control optimization. However, the widespread use of approximation methods is hampered by the lack of the ability to approximate a complex simulation model which characterizes the dynamic feature with multiple inputs and multiple outputs (MIMO) in a large domain. In this paper, a novel global approximation method for simulation models based on the RBF response surface set is proposed. Firstly, incremental building technique of RBF response surface set was studied, and was applied to approximate MIMO models. Several mathematical tests were presented to demonstrate the feasibility and effectiveness of the technique. Secondly, the approximation for complex simulation models, especially for dynamic models with state variables, was addressed. A simple test was given to illustrate the approximation process and effectiveness of a simulation model. Lastly, as an engineering application, the proposed method was utilized to approximate the power-train of a pure electric vehicle, and the approximation model was successfully applied in real-time simulation platform.

Keywords: Multi-dimensional global approximation, blackbox function, response surface set, real-time simulation, multiple inputs and multiple outputs.

1 Introduction

It is quite applicable and useful to substitute a complex nonlinear black-box function (i.e., original model) with another simple model under some special occasions, such as parametric experiment, sensibility analysis, real-time simulation, and design or control optimization for the original model. For a typical example, the

[1] National CAD supported software engineering centre in Huazhong University of Science and Technology, Wuhan, P.R.China, 430074.

[2] Corresponding author. E-mail: cad.wyz @hust.edu.cn

[3] Institute of Dongguan-Sun Yat-Sen University.

approximation for an original finite-element analysis model can greatly speed up the process of a parametric experiment or design optimization for the model. In the real-time or HIL (hardware in loop) simulation field, if an original simulation model is too complex to be solved in real-time steps, the relationship from the inputs to the outputs of the component model can be approximated by a simple substitute model to realize real-time or HIL simulation. Approximating the complex model is an effective method to solve this type of problem.

With the development of approximating technologies, there are now some commonly used methods that can mainly be classified into two classes: look-up table and response surface method.

Look-up table is used to interpolate among samples to give an approximate value for a black-box function. However, it is only suitable for lower-dimensional problems, and it is difficult to build a look-up table with appropriate data grids in the global domain of the original model.

Response surface method (RSM) is one of the most widely used global approximation tools. Generally, a response surface for an original model is also known as a surrogate or a meta-model. To this end, researchers on RSM were concentrated on several key techniques [Wang and Shan (2007)]: PRS (polynomial response surface) [Box and Wilson (1951)], Kriging [Matheron (1963); Sacks, Welch, and Mitchell (1989)], RBF (radial basis functions) [Hardy (1971); Buhmann (2000); Elgohary, Dong, Junkins, and Atluri (2014); Elgohary, Dong, Junkins, and Atluri (2014)], SVR (support vector regression) [Vapnik, Golowich, and Smola (1997); Prakasvudhisarn, Trafalis, and Raman (2003)], and MARS (multivariate adaptive regression spline) [Friedman (1991)]. Among the five RSM techniques, PRS, SVR and MARS belong to regression methods, while Kriging and RBF are considered interpolation methods. The PRS technique is particularly suitable for lower order functions and low-to-medium dimensional problems near a local area. Although some adaptive response surface methods are proposed, see [Simpson, Landman, and Giroux (2008); Nguyen, Sellier, and Duprat (2009)], they perform poorly for high-order functions and high nonlinear models upon a global large domain. According to S. M. Clarke et al., see [Clarke, Griebsch, and Simpson (2005)], as a regression technique, SVR had the best overall performance for the test bed of 26 engineering analysis functions in comparison to other approximating techniques. MARS takes the form of an expansion in multivariate spline basis functions. As a regression technique, MARS is suitable for multi-dimensional global approximation problems. However, according to Friedman, see [Friedman (1991)], the MARS algorithm involves the calculation of a lack-of-fit procedure for finding the optimal knots vector, which is very time-consuming. In addition, the recursive partitioning strategies like classification and regression trees is lack of continuity, which affects model accuracy; the recursive partitioning often results in a poor predictive ability for even low-order performance functions when new data are introduced. The Kriging technique is suitable for high-order functions and low dimensional problems, but it is less efficient for low order functions and high dimensional problems, see [Gu, Li, and Dong (2012)].

To approximate a deterministic computer simulation model, the interpolation technique is preferred to the regression one because the regression model is not sensitive to the sampling data, which may lead to large errors between the real and the approximate data. Therefore, the PRS, SVR and MARS techniques are not suitable to approximate a computer model. RBF and Kriging are both interpolated metamodels for nonlinear functions, but Kriging is not suitable for high dimensional problems. In addition to the accuracy, RBF is selected as approximate model because of it fast and rapid evaluation, see [Beatson and Newsam (1992); Roussos and Baxter (2005)]. RBF technique performs well for high dimensional and high-order nonlinear problems. Although it is less efficient in dealing with linear functions, this can be conquered by augmenting with a linear polynomial item, see [Fang and Horstemeyer (2006)]. Among the literature, there have already been global approximation methods using the RBF method. In [Fang and Horstemeyer (2006)], a variety of existing RBF basis functions was compared in both non-augmented and augmented forms with various types of responses and limited numbers of samples. However, they were all constructed in one-stage sampling and did not fit for approximation of black-box model. In [Driscoll and Heryudono (2007)], a new adaptive algorithm was constructed for RBF method, which was applied in interpolation, boundary-value, and initial-boundary-value problems with localized features. However, the algorithm focused only on the boundary interpolation of the function with a non-cuboid domain. In [Iske and Levesley (2005)], a method was studied for multilevel scattered data approximation by using compactly supported radial basis functions with adaptive domain decomposition. Nevertheless, the method is unsuitable for the approximation of a black-box model that provides data through expensive calculating.

Generally, a black-box function, such as a simulation model, always has multiple inputs and multiple outputs (MIMO). However, traditional RSM techniques are only used for approximating a model with a single output. These methods are not suitable for dealing with a MIMO model because it needs to approximate each output with a single response. Not only will this method cause a lot of memory consumption, but it will also reduce the efficiency of the construction and evaluation of the response. Then, some new techniques appeared. The RBF neural networks was used to approximate the unknown nonlinear functions of MIMO control systems, see [Park, Venayagamoorthy, and Harley (2005); Zhen, Qi, and Li (2014)].

In [Peng, Wu, and Inoussa (2009); Peng, Kitagawa, and Wu (2011); Wu, Peng, and Chen (2014)], The Peng, H. et al. proposed a MIMO RBF-ARX model for nonlinear model-based predictive control (NMPC), and provided a state-space form of NMPC. The predictive control strategy for nonlinear system is proven to be effective.

In this paper, a new global approximation method based on RBF response surface set is proposed. This method can address the problem for black-box model with MIMO efficiently and can be used in many occasions, such as parametric experiment, sensibility analysis, etc. Here, RBF is adopted because it is a type of interpolation response surface method, which can be easily extended to approximate multiple outputs. In addition, the RBF response surface set can be updated incrementally and efficiently during the sequential sampling and refining process. To approximate a complex simulation model using the proposed method, the whole process is put forward. Additionally, if the simulation model has the properties of state variables, the dynamic features of the simulation model also need to be considered. In this paper, a method that can handle the state variable of the simulation model is proposed. It should be noted that if the final approximate model constructed based on the simulation model meets the requirement of accuracy, it can be reused in other applications.

The remainder of this paper is organized as follows: In Section2, the theory of the RBF method is briefly reviewed. In Section3, the method of the RBF-based response surface set is proposed, and the build process is introduced. The proposed method is tested using several mathematical tests. In Section4, the process to approximate the simulation model is put forward, and its validity is verified by a simple simulation model. In Section5, as an engineering application, the powertrain of a pure electric vehicle simulation model, is approximated based on the proposed methods, and the approximation model is applied in real-time simulation platform. Finally, the conclusions of this work are given in Section 6.

2 A review of the RBF method

Through statistically designed experiments, the sampling data S: $[x_{i,}y_i]$ (i =1, 2… n) can be obtained, where $\boldsymbol{x}_i$is a p-dimensional vector (i.e., sampling site), and $\boldsymbol{y}_i$ is its corresponding real response value. Using the sampling data, the RBF model can be constructed in the general form of

$$\hat{y} = \hat{f}(\boldsymbol{x}) = \sum_{i=1}^{n} \lambda_i \varphi(\|\boldsymbol{x} - \boldsymbol{x}_i\|) = \boldsymbol{\Phi} \cdot \boldsymbol{\lambda} \tag{1}$$

where n is the number of sampling points, $\boldsymbol{x}$ is a vector of design variables, $\boldsymbol{x}_i$ is a vector value of the design variables at the i^{th} sampling point, ϕ is a basis function,

$||\boldsymbol{x}$-$\boldsymbol{x}_i||$ is the Euclidean norm, and $\boldsymbol{\lambda}_i$ is the coefficient for the i^{th} basis function, $\boldsymbol{\Phi}$=[ϕ_1, ϕ_2,..., ϕ_N] (ϕ_j=$\phi||\boldsymbol{x}$ -$\boldsymbol{x}_i||$), $\boldsymbol{\lambda}$=[λ_1,λ_2,...,$\lambda_N]^T$. The approximation function y is a linear combination of some RBFs with weight coefficients.

Replacing the $\boldsymbol{x}$ and y in equation (1) with the sampling data $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$, we can determine the N coefficients $\boldsymbol{\lambda}_i$ through solving the following linear equations:

$$\boldsymbol{\lambda} = A^{-1}\boldsymbol{y} \tag{2}$$

where A is the design matrix, A_{ij}=$(||\boldsymbol{x}_i - -\boldsymbol{x}_j||)$, and $\boldsymbol{y}$ is the vector: y=[y_1,y_2,...,$y_n]^T$.

There are several types of radial functions, such as cubic, linear, Gaussian, multi-quadrics, inverse-multi-quadric, thin-plate spline and logistic functions, which are shown in Table. 1. In this article, the linear function is adopted.

Although the RBF is good for high-order nonlinear models, it has been shown to be inappropriate for linear models. To overcome this shortcoming, a linear polynomial is added to the RBF model. The augmented RBF, see [Fang and Horstemeyer (2006)], is shown as

$$\hat{f}(\boldsymbol{x}) = \sum_{i=1}^{n} \lambda_i \varphi(||\boldsymbol{x} - \boldsymbol{x}_i||) + \sum_{j=1}^{p} c_j g_j(\boldsymbol{x}) \tag{3}$$

where $g(\boldsymbol{x})$ is a linear polynomial function, p is the total number of terms in the polynomial, and c_j(j=1,2,3,...,p) are the unknown coefficients. However, the equation (3) is underdetermined, as the equations created with available data points is less than the number of parameters to be solved. Therefore, the orthogonality condition is needed to be imposed on coefficients λ so that

$$\sum_{j=1}^{n} \lambda_i g_j(\boldsymbol{x}_i) = 0 \quad j = 1,2,3.....,p \tag{4}$$

Combining equations (3) and (4), $(n+p)$ equations are obtained and the matrix form is given as

$$\begin{pmatrix} A\,G \\ G^T\,0 \end{pmatrix} \begin{pmatrix} \lambda \\ c \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix} \tag{5}$$

where $G_{i,j}$=g$_j$($\boldsymbol{x}_i$) (i=1,2,..., n, j=1,2,..., p) and c=[c_1,c_2,..., $c_p]^T$. Solving equation (5) gives coefficients λ and c for the augmented RBF function given in equation (3).

Additionally, this paper will not discuss the different forms of the RBF model, which means that the proposed method can be used for all types of the RBF models.

Table 1: Commonly-used basis functions.

Commonly-used basis functions	Function
Linear	$\varphi(r) = r$
Cubic	$\varphi(r) = (r+c)^3$
Thin-plate spline	$\varphi(r) = r^2 \ln(cr)$
Gaussian	$\varphi(r) = e^{-cr^2}$
Multi-quadric	$\varphi(r) = \sqrt{r^2+c^2}$
Inverse-multi-quadric	$\varphi(r) = \frac{1}{\sqrt{r^2+c^2}}$

where c is a constant, and $0 < c \leq 1$.

3 RBF-based global approximation method for black-box functions

3.1 RBF-based response surface set method

For an original model with m outputs and p inputs or dimensions, described as:

$$\boldsymbol{y} = f(\boldsymbol{x}) \tag{6}$$

where $\boldsymbol{y}$: $\Omega \subset \Re^m$, $\boldsymbol{x}$: $\Omega \subset \Re^p$; and $f : \Re^p \to \Re^m$. As shown in Figure.1, the original black-box function in (1) can be approximated as:

$$\hat{\boldsymbol{y}} = \hat{f}(\boldsymbol{x}) \tag{7}$$

Traditionally, the RSM techniques are only used for approximating a model with a single output. It is single-valued mapping from a sampling point set S to the real response value $\boldsymbol{Y}$ ($\boldsymbol{Y}$ is a vector). As shown in Figure. 2 (a), for any single sampling point x_i (the dimension of $x_i$is the number of independent variables or inputs), there exists a corresponding response scalar value y_i.

To address the MIMO model, the common approach is to construct each output of MIMO model by RSM method, respectively, and group these approximate models together. The adaptive sampling strategies can be used to sample data for a higher precision approximate model.

$$\begin{array}{l} \hat{y}_1 = \hat{f}_1(\boldsymbol{x}_1) \\ \hat{y}_2 = \hat{f}_2(\boldsymbol{x}_2) \\ \ldots\ldots \\ \hat{y}_m = \hat{f}_m(\boldsymbol{x}_m) \end{array} \quad \text{for} \quad \boldsymbol{x} = (x1, x2, \ldots, xp) \tag{8}$$

The constructive process of these approximate models is a time-consuming. Take an example of two outputs, if the approximate model y_1 is constructed by RSM method using a certain amount of sampling data S_0 through expensive calculation. To save time, the sampling data S_0 is directly used in the next approximate model y_2. Then y_2 is constructed using S_0 and additional sampling data S_1 which is used to improve the accuracy for y_2. However, the additional sampling data S_1 is also needed to reconstruct the y_1 to improve the accuracy of the approximate model. So it is complicated to construct the approximate models for multiple outputs.

Generally, a black-box model, such as the simulation model, always has multiple outputs. In fact, some RSM techniques can be easily extended to approximate multiple outputs. Now, we introduce the so-called response surface set (RSS), based on the RBF model. RSS is a multi-value mapping from the sampling point set to the real response value $\boldsymbol{Y}$ ($\boldsymbol{Y}$ is a matrix). For any single sample point $\boldsymbol{x}_i$, there is a corresponding vector $\boldsymbol{y}_i$ (the dimension of $\boldsymbol{y}_i$ is the number of dependent variables or outputs), as shown in Figure. 2 (b). The RSS will extend the traditional single response to multiple response RSM techniques.

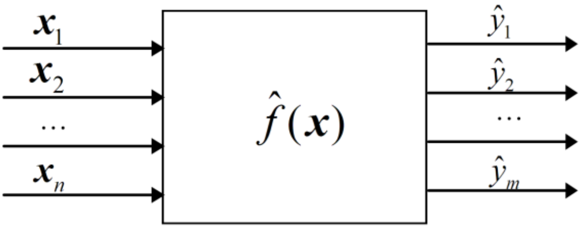
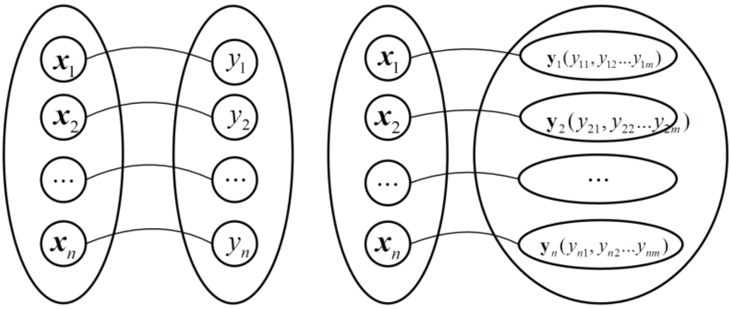


Figure 1: The approximate model of the original model.


$\boldsymbol{x}_1$
$\boldsymbol{x}_2$
…
$\boldsymbol{x}_n$
y_1
y_2
…
y_n
$\boldsymbol{x}_1$
$\boldsymbol{x}_2$
…
$\boldsymbol{x}_n$
$\mathbf{y}_1(y_{11}, y_{12} \cdots y_{1m})$
$\mathbf{y}_2(y_{21}, y_{22} \cdots y_{2m})$
…
$\mathbf{y}_n(y_{n1}, y_{n2} \cdots y_{nm})$

(a) Single-valued response surface (b) Response surfaces set

Figure 2: Single-valued response surface and multi-valued response surfaces set.

When using augmented RBF models, the multiple response surfaces are constructed according to equation (3):

$$\begin{aligned}
&\hat{\boldsymbol{y}}^1 = \hat{f}^1(\boldsymbol{x}) = \sum_{i=1}^{n} \lambda_i^1 \varphi(\|\boldsymbol{x}-\boldsymbol{x}_i\|) + \sum_{j=1}^{p} c_j^1 g_j(\boldsymbol{x}) \\
&\hat{\boldsymbol{y}}^2 = \hat{f}^2(\boldsymbol{x}) = \sum_{i=1}^{n} \lambda_i^2 \varphi(\|\boldsymbol{x}-\boldsymbol{x}_i\|) + \sum_{j=1}^{p} c_j^2 g_j(\boldsymbol{x}) \\
&\ldots\ldots \\
&\hat{\boldsymbol{y}}^m = \hat{f}^m(\boldsymbol{x}) = \sum_{i=1}^{n} \lambda_i^m \varphi(\|\boldsymbol{x}-\boldsymbol{x}_i\|) + \sum_{j=1}^{p} c_j^m g_j(\boldsymbol{x})
\end{aligned} \tag{9}$$

and the orthogonality condition is also needed to imposed on coefficients λ so that

$$\left.\begin{aligned}
&\sum_{j=1}^{n} \lambda_i^1 g_j(\boldsymbol{x}_i) = 0 \\
&\sum_{j=1}^{n} \lambda_i^2 g_j(\boldsymbol{x}_i) = 0 \\
&\ldots\ldots \\
&\sum_{j=1}^{n} \lambda_i^m g_j(\boldsymbol{x}_i) = 0
\end{aligned}\right. \quad \text{for } j = 1,2,3\ldots..,p \tag{10}$$

According to the n sampling points $[\boldsymbol{x}_i, \boldsymbol{y}_i]$ (i=1,2,...,n), here, $\boldsymbol{y}_i$ is a vector of the m outputs, and we can obtain all of the coefficients of the m augmented RBF models by solving the following equation,

$$\boldsymbol{Y} = A\boldsymbol{\Lambda} + B\Delta \tag{11}$$

and the orthogonality condition

$$B^T\ \boldsymbol{\Lambda} = 0 \tag{12}$$

where $\boldsymbol{Y} = [\boldsymbol{y}^1, \boldsymbol{y}^2, ..., \boldsymbol{y}^m]$; $\boldsymbol{\Lambda} = [\boldsymbol{\lambda}^1, \boldsymbol{\lambda}^2, ..., \boldsymbol{\lambda}^m]$; both Λ and $\boldsymbol{Y}$ are $n \times m$ matrix; $\lambda^j (j = 1,2,\ldots,m)$ is a vector; the element of matrix $\boldsymbol{y}_{ij}$ is the j^{th} output of the i^{th} sampling point; B is $n \times p$ matrix; the element of matrix B_{ij} is the monomial terms in the polynomial; Δ is $p \times m$ matrix; and the element of matrix Δ_{ij} is coefficients of polynomial.

Combining equations (11) and (12), $m(n+p)$ equations can be obtained and the matrix form is given as

$$\begin{pmatrix} A_{n\times n}\ B_{n\times p} \\ B^T_{p\times n}\ 0_{p\times p} \end{pmatrix} \begin{pmatrix} \boldsymbol{\Lambda}_{n\times m} \\ \Delta_{p\times m} \end{pmatrix} = \begin{pmatrix} \boldsymbol{Y}_{n\times m} \\ 0_{p\times m} \end{pmatrix} \tag{13}$$

Solving equation (13) gives coefficients $\mathbf{\Lambda}$ and Δ for the augmented RBF function given in equation (6).

In fact, all of the augmented RBF models of the RSS share the same design matrix A; therefore, it is much more efficient and compact to approximate multiple responses rather than using multiple independent response surfaces.

3.2 Incremental construction of RSS

The equation (13) can also be expressed as

$$\begin{pmatrix} 0_{p\times p} \; B^T_{p\times n} \\ B_{n\times p} \; A_{n\times n} \end{pmatrix} \begin{pmatrix} \Delta_{p\times m} \\ \mathbf{\Lambda}_{n\times m} \end{pmatrix} = \begin{pmatrix} 0_{p\times m} \\ \boldsymbol{Y}_{n\times m} \end{pmatrix} \tag{14}$$

For an existing RSS model y, if we add another q sampling points S′: [x'_i, y'_i] (i=1, 2, …, q), the matrix form is given as

$$\begin{pmatrix} 0_{p\times p} \; B^T_{p\times n} \; Q^T_{p\times q} \\ B_{n\times p} \; A_{n\times n} \; V^T_{n\times q} \\ Q_{q\times p} \; V_{q\times n} \; D_{q\times q} \end{pmatrix} \begin{pmatrix} \Delta_{p\times m} \\ \begin{pmatrix} \mathbf{\Lambda}_{n\times m} \\ \beta_{q\times m} \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 0_{p\times m} \\ \boldsymbol{Y}_{n\times m} \\ \alpha_{m\times m} \end{pmatrix} \tag{15}$$

where $V_{ij} = \varphi(\left\|\boldsymbol{x}_i - \boldsymbol{x}'_j\right\|)$, $D_{ij} = \varphi(\left\|\boldsymbol{x}'_i - \boldsymbol{x}'_j\right\|)$. Let $A = \begin{pmatrix} 0_{p\times p} \; B^T_{p\times n} \\ B_{n\times p} \; A_{n\times n} \end{pmatrix}$, $U = \begin{pmatrix} Q^T_{p\times q} \\ V^T_{n\times q} \end{pmatrix}$, $V = (\, Q_{q\times p} \; V_{q\times n})$, $\boldsymbol{\mu} = \begin{pmatrix} \Delta_{p\times m} \\ \mathbf{\Lambda}_{n\times m} \end{pmatrix}$, $U = \begin{pmatrix} 0_{p\times m} \\ \boldsymbol{Y}_{n\times m} \end{pmatrix}$, $D = (D_{q\times q})$, $\boldsymbol{\delta} = (\beta_{q\times m})$, and $\boldsymbol{U}' = (\alpha_{m\times m})$. The matrix form in equation (15) can be expressed as

$$\begin{pmatrix} A \; U \\ V \; D \end{pmatrix} \begin{pmatrix} \boldsymbol{\mu} \\ \boldsymbol{\delta} \end{pmatrix} = \begin{pmatrix} U \\ U' \end{pmatrix} \quad => \quad \begin{pmatrix} \boldsymbol{\mu} \\ \boldsymbol{\delta} \end{pmatrix} = A_1^{-1} \begin{pmatrix} U \\ U' \end{pmatrix} \tag{16}$$

where $A_1 = \begin{pmatrix} A \; U \\ V \; D \end{pmatrix}$. The coefficients of new RSS model $\boldsymbol{\mu}$ and $\boldsymbol{\delta}$ can be obtained through solving the

inverse matrix A^{-1}. According to Duncan's study on the inverse of a block matrix [27], we have

$$A_1^{-1} M^{-1} = \begin{bmatrix} A^{-1} + A^{-1}UC^{-1}VA^{-1} & -A^{-1}UC^{-1} \\ -C^{-1}VA^{-1} & C^{-1} \end{bmatrix} \tag{17}$$

where $V = U^T, C = D - V^{-1}AU$.

As we know, the complexity for calculation A_1^{-1} is $O((N+K)^3)$ originally, but if we store the A_1^{-1} at the previous step, the complexity then becomes $O(K^3)$ which is mainly consumed for calculating C^{-1}.

3.3 The construction process of the RBF-based response surface set

The RBF-based response surface set is constructed based on the Multi-dimensional Global Approximation (MGA) algorithm, see [Wei, Wu, and Chen (2012)], and the procedure of the MGA algorithm for multiple outputs is similar to that for a single output. The algorithm can be summarized in six steps as follows:

Step 1: Initial sampling. To ensure a unique construction process of RSS, Grid sampling with identical level q is adopted in the initial design stage. The points generated by Grid sampling are called "expensive points" because their values are evaluated by black-box simulation functions or analytical models.

Step 2: Construct the initial RSS model. The first modification involves the construction of the initial RSS model, the evaluation using RSS and the incremental construction of RSS rather than the single RBF model. These have been described in Section 3.1.

Step 3: Check stopping criteria. The off-design test, such as Root-mean-square errors (RMSE), is not suitable for stopping criteria, as too many function evaluations need to be done. In this paper, two criteria are adopted at this step: one is the maximum number of iterations, M1; the other is the maximum number of continuous relative error M2 between the corresponding true function value and function value calculated from the RBF models. The former limits the number of function evaluations while the latter means that the model is accurate enough. The continuous relative error is calculated as

$$|y_i - \hat{y}_{i-1}| \leq \delta \tag{18}$$

where y_i is the real response value of a sampling site $\boldsymbol{x}_i$; $\hat{y}_{i-1}$ is the approximate value of the RBF model constructed without the point [$\boldsymbol{x}_i$, y_i]; δ is the absolute tolerance, a positive number set by user.

If one of the criteria is satisfied, go to ***Step6***.

Step 4: Sequential Optimal Sampling. The Sequential Optimal Sampling (SOS) is a typical global optimization problem and can be settled by DIRECT algorithm, see [Jones (2001)]. The criterion of SOS [Wei, Wu, and Chen (2012)] has the form of

$$\begin{cases} find : \boldsymbol{x} \\ \max : K(\boldsymbol{x}) \cdot d_{\min}^{D}(\boldsymbol{x}) \end{cases} \tag{19}$$

where $d_{min}(\boldsymbol{x})$ is minimum distance from $\boldsymbol{x}$ to other existing sampling sites, $K(\boldsymbol{x})$ is the curvature of the site $\boldsymbol{x}$ on the response surface, and the power $D$is set to 2.

The calculation for curvature $K(x)$ should be modified to meet the properties of the RSS method. The curvature at a given point on the RSS model can be calculated by using:

$$K(\boldsymbol{x}) = \max(K_i(\boldsymbol{x})) \tag{20}$$

where $K_i(\boldsymbol{x})$ (i=1,2,...,m) is the curvature of the i^{th} response surface.

Step 5: Update the RSS model. After adding a new sampling point (optimal site) to the current sampling point set, the RSS model should be constructed incrementally rather than recalculating the RSS model completely. To realize its incremental construction, the sampling sites S and the inverse matrix of design matrix A^{-1} should be contained in the RSS model structure. These have been described in Section 3.2. Go to ***Step3***.

Step 6: Stop.

3.4 Mathematical test

In this section, the performance of the MGA algorithm, *RMSE*, (through additional off-design testing), will be compared with some mathematical models. These mathematical models are mainly composed of some well-known benchmark functions. All of these tested models are approximated by RBF-based RSS methods, but the response surfaces are constructed through three sampling strategies (i.e., LHD, Grid, and SOS) with the same sampling size. Note that the LHD and Grid sampling strategies are common used sampling method, and can spatial sample in the whole design space, just like the SOS method.

Besides, to compare the performances of RBF-based RSS model, the Kriging-based RSS model is also constructed using corresponding incremental construction method. Both of these two models adopt the SOS sampling method.

The *RMSE* is expressed as

$$RMSE = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(f(x^k) - f_{app}(x^k))^2} \tag{21}$$

Where n is the number of test points, $f(x^k)$ represents the actual function value at the test point x^k, and $f_{app}(x^k)$ is the value estimated by the surrogate at the test point x^k.

Note that the *RMSE* of the RSS model is calculated via the average one:

$$\overline{RMSE} = \frac{\sum_{i=1}^{m} RMSE_i}{m} \tag{22}$$

where $RMSE_i$ (i=1,2,...,m) is the *RMSE* of the i^{th} response surface.

Example 1: An expanded form of the Easom function with 2 inputs and 2 outputs: 2-dimension

$$\begin{aligned} y_1 &= -\cos x_1 \cos x_2 e^{-(x_1-\pi)^2-(x_2-\pi)^2} \\ y_2 &= -\cos x_1 \cos x_2 e^{-(x_1+\pi)^2-(x_2+\pi)^2} \end{aligned} \tag{23}$$

Example 2: This model is a hybrid of the Golden-Price function (GP), a simple exponential function and the Banana Function with 2 inputs and 3 outputs: 2-dimension

$$\begin{aligned} y_1 &= 30+1/200000(1+(x_1+x_2)^2(19-14x_1+3x_1^2-14x_2+6x_1x_2+3x_2^2))\ldots \\ &\quad (30+(2x_1-3x_2^2)(18-32x_1+12x_1^2+48x_2-36x_1x_2+27x_2^2)) \\ y_2 &= 60+x_1e^{x_2}+x_2e^{x_1} \\ y_3 &= 1/100(100(x_2-x_1^2)^2+(1-x_1)^2) \end{aligned} \tag{24}$$

Example 3: This model is a hybrid of the Colville function, Levy function, Dixon-Price Function, Trid function, Rosenbrock function, and Zakharov function with 4 inputs and 6 outputs: 4-dimension

$$\begin{aligned} y_1 &= 1/10^5(100(x_1^2\text{-}x_2)^2\text{+}(x_1\text{-}1)^2\text{+}(x_3\text{-}1)^2\text{+}90(x_3^2\text{-}x_4)^2 \\ &\quad \text{+}10.1((x_2\text{-}1)^2\text{+}(x_4\text{-}1)^2)\text{+}19.8(x_2\text{-}1)(x_4\text{-}1)) \\ y_2 &= 1/10(\sin^2(\pi w_1) \\ &\quad +\sum_{i=1}^{3}(w_i-1)^2[1+10\sin^2(\pi w_i+1)]+(w_d-1)^2[1+\sin^2(2\pi w_d)]) \\ y_3 &= 1/10^4((x_1-1)^2+\sum_{i=2}^{4} i(2x_i^2-x_{i-1})^2) \\ y_4 &= 1/10(\sum_{i=1}^{4}(x_i-1)^2-\sum_{i=2}^{4}x_ix_{i-1}) \\ y_5 &= 1/10^5(\sum_{i=1}^{3}[100(x_{i+1}-x_i^2)^2+(x_i-1)^2]) \\ y_6 &= 1/10^5(\sum_{i=1}^{4}x_i^2+(\sum_{i=1}^{4}0.5ix_i)^2+(\sum_{i=1}^{4}0.5ix_i)^4) \end{aligned} \tag{25}$$

where $w_i = 1+\frac{x_i-1}{4}, i = 1,\ldots,d.$

Example 4: This model is a hybrid of the Hartman function (HN), Zakharov function, Trid function, and Rosenbrock functions with 6 inputs and 4 outputs: 6-dimension

$$
\begin{aligned}
y_1 &= -\sum_{i=1}^{4} c_i \exp[-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2] \\
y_2 &= 1/10^7(\sum_{i=1}^{6} x_i^2 + (\sum_{i=1}^{6} 0.5ix_i)^2 + (\sum_{i=1}^{6} 0.5ix_i)^4) \\
y_3 &= 1/10(\sum_{i=1}^{6} (x_i - 1)^2 - \sum_{i=2}^{6} x_i x_{i-1}) \\
y_4 &= 1/10^5(\sum_{i=1}^{5} [(1 - x_i)^2 + 100(x_{i+1} - x_i^2)^2])
\end{aligned}
\tag{26}
$$

where c=[1,1.2,3,3.2]^T,

$$
a = \begin{bmatrix} 10\ 3\ 17\ 3.05\ 17\ 8 \\ 0.05\ 10\ 17\ 0.1\ 8\ 14 \\ 3\ 3.5\ 1.7\ 10\ 17\ 8 \\ 17\ 8\ 0.05\ 10\ 0.1\ 14 \end{bmatrix}, \quad p = 10^{-4} \begin{bmatrix} 1312\ 1696\ 5569\ 124\ 8283\ 5886 \\ 2329\ 4135\ 8307\ 3736\ 1001\ 9991 \\ 2348\ 1451\ 3522\ 2883\ 3047\ 6650 \\ 4047\ 8828\ 8732\ 5743\ 1091\ 381 \end{bmatrix}
$$

Example 5: This model is a hybrid of the Zakharov function, Trid function, and Rosenbrock functions with 10 inputs

and 3 outputs: 10-dimension

$$
\begin{aligned}
y_1 &= 1/10^8(\sum_{i=1}^{10} x_i^2 + (\sum_{i=1}^{10} 0.5ix_i)^2 + (\sum_{i=1}^{10} 0.5ix_i)^4) \\
y_2 &= 1/10(\sum_{i=1}^{10} (x_i - 1)^2 - \sum_{i=2}^{10} x_i x_{i-1}) \\
y_3 &= 1/10^5(\sum_{i=1}^{9} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2])
\end{aligned}
\tag{27}
$$

Example 6: This model is a hybrid of the Dixon-Price Function, and Zakharov Function with 13 inputs and 2 outputs:

13-dimension

$$
\begin{aligned}
y_1 &= 1/10^5((x_1 - 1)^2 + \sum_{i=2}^{13} i(2x_i^2 - x_{i-1})^2) \\
y_2 &= 1/10^9(\sum_{i=1}^{13} x_i^2 + (\sum_{i=1}^{13} 0.5ix_i)^2 + (\sum_{i=1}^{13} 0.5ix_i)^4)
\end{aligned}
\tag{28}
$$

The parameters of D, q, M1, M2, and δ in Table. 2 represent the smoothness of the approximate model surface, sampling number, the maximum number of iterations, maximum number of continuous relative error between the corresponding true function value and function value calculated from the RBF models, and the absolute tolerance. The performances about *RMSE* are compared in Table. 3, and some of the approximate models are shown in Figure. 3 and Figure. 4. As observed from Table. 3, the SOS method can address the multiple outputs quite well, and the *RMSE* of the SOS method is typically smaller than the other methods. In a word, the SOS method can sample in a quickly and better way, and the approximate model constructed using these sampling points has a higher precision than those of LHD and Grid method.

Besides, from the planer view of Figure. 3, it can be found that the sample points gained from the SOS method, unlike GRID and LHD, gather in a region that has the feature of larger curvature than other places and smaller distances from the new sample point to other existing sampling sites.

Moreover, the results of the comparison between RBF-based RSS model and Kriging-based RSS model are also shown in Table. 4. As observed form Table. 4, the *RMSE* of RBF model is close to the one obtained by Kriging model, however, the time it takes to construct the Kriging model is much longer than RBF model, particularly when the problem is a complicated one.

Table 2: Preprocessing results.

Example	Space	D	q	M1	M2	δ
1	$x_1,x_2 \in [-10,\ 10]$	2	3	200	5	[1e-2,1e-2]
2	$x_1,x_2 \in [-2,\ 2]$	2	3	200	5	[1e-2,1e-2,1e-2]
3	$x_1,...,x_4 \in [-10,\ 10]$	2	3	400	5	[1e-2, 1e-2, 1e-2, 1e-2, 1e-2, 1e-2]
4	$x_1,...,x_6 \in [-10,\ 10]$	2	3	500	5	[1e-2, 1e-2, 1e-2, 1e-2]
5	$x_1,...,x_{10} \in [-10,\ 10]$	2	2	900	5	[1e-2,1e-2, 1e-2]
6	$x_1,...,x_{13} \in [-10,\ 10]$	2	2	1000	5	[1e-2, 1e-2]

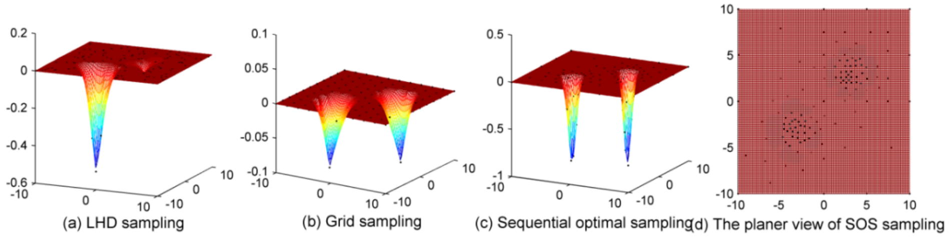
4 Global approximation method for the simulation model

The simulation model (as the original model) in many fields, such as parametric experiment, sensibility analysis, real-time simulation, and design or control optimization, is always a complex nonlinear model. To test the simulation model in the real-time simulation or HIL, simplifying or approximating is necessary.

Several steps should be taken to address the simulation model before obtaining the approximate model.

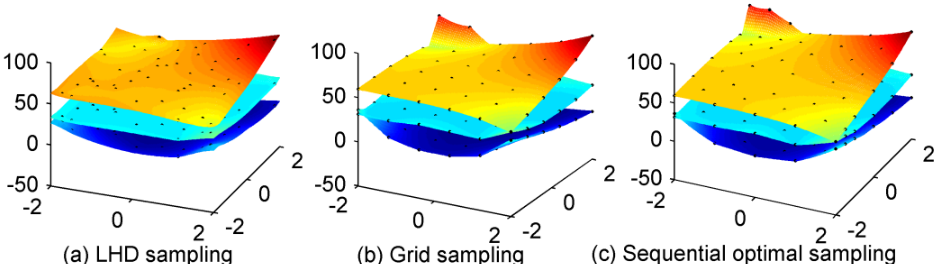
Table 3: Test results.

Example	$\overline{RMSE}$		
	LHD	Grid	SOS
1	0.0358	0.0416	0.0138
2	0.3134	0.1257	0.0827
3	1.0701	2.3294	0.8110
4	0.1022	1.4352	0.6205
5	1.9374	11.2347	0.7185
6	0.7889	9.1854	0.4155



(a) LHD sampling (b) Grid sampling (c) Sequential optimal sampling (d) The planer view of SOS sampling

Figure 3: RBF-based RSS for the model in (23) by using different sampling strategies.



(a) LHD sampling (b) Grid sampling (c) Sequential optimal sampling

Figure 4: RBF-based RSS for the model in (24) by using different sampling strategies.

Table 4: Test results for different RSM RSS model.

Example	RBF-based RSS model		Kriging-based RSS model	
	$\overline{RMSE}$	Time/s	$\overline{RMSE}$	Time/s
1	0.0138	107.4494	0.0121	66.2585
2	0.0827	165.6150	0.0365	280.6364
3	0.8110	3.8511e+003	0.9257	5.4121e+003
4	0.6205	1.9513e+004	0.2205	3.2478e+004
5	0.7185	1.0543e+005	0.5332	1.9755e+005
6	0.4155	1.1725e+006	0.5065	2.734e+006

4.1 The process to approximate a simulation component

As shown in Figure. 5, the simulation model has three modules: the previous module, next module and current module. Ignoring the previous and the next modules, we only discuss the approximate method for the current module.

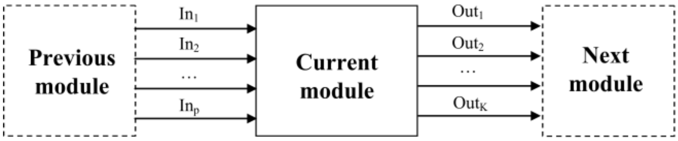


Figure 5: Simulation module with multi-inputs and outputs.

Step 1: Establish the initial ISM model. Before approximating the "current module", an independent simulation module (ISM) model of the current module needs to be established. The ISM model is a black-box function that can be simulated independently. Several pretreatment sub-steps need to be adopted to establish the ISM model:

- Deal with the inputs and outputs of the module. In general, there are not explicit inputs and outputs in the module, and it is necessary to address the inputs and outputs (including the state variables added) from the simulation model according to the requirement.
- Analyze the scope of input variables of the module. According to the simulation history data or the physical analysis of the simulation model, it is not difficult to get the scope of all of the input variables (including the state vari-

ables added) of the module. All of the sampling sites will be restricted in this scope.

- Deal with the time-step t0. In fact, the current module is time-dependent, which means that with different simulation time t the outputs (including the state variables added) will be different. As we know, a constant time-step (such as t0 = 0.1 second) is normally adopted when running the real-time simulation. The same case happens in the control optimization where the time-step is usually set to 1 second (t0 = 1), see [Liu (2007)]. Note that the model will be run with the time-step t0 to evaluate the outputs from the different sampling sites.

To this end, the Independent simulation module for an original model component can be expressed as Figure.6 shows. It has three facts: inputs and outputs as the original component (In_1, In_2, ..., In_p, Out_1, Out_2, ..., Out_k), the inputs and outputs of state variables (S_1, S_2,..., S_M, $S_1\prime$, $S_2\prime$, ..., $S_M\prime$), and the time-step constant t_0. Dealing with the state variables will be discussed in Section 4.2.

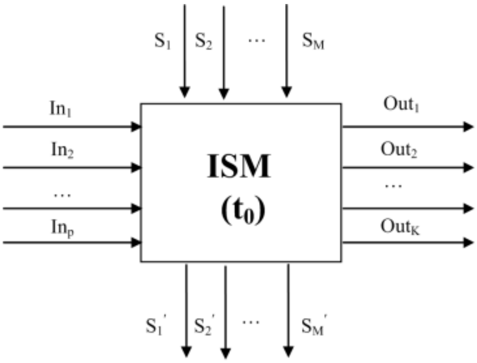


Figure 6: Independent simulation module of the original component.

Step 2: Construct the approximate model. Based on the Independent simulation module, the approximate model of original component can be constructed according to the RBF-based RSS method mentioned in section 3.3. Note that the Independent simulation module will be run with the time step t_0 in each iteration to evaluate the outputs from the different sampling sites.

Step 3: Check the accuracy of the approximate model. The accuracy of the approximate model should be verified compared with the original simulation model. Then the component needed to be approximated in the original simulation model can be replaced with approximate model. The properties of the simulation model with approximate component compared with the original simulation model can be checked during the test, such as whether the values of some interested variables are close enough to the values of the original simulation model. Additionally, the accuracy of the approximate model can be measured by R^2 in (31) through off-design testing.

If the accuracy of the approximate model meets certain requirements, go to ***Step 5*** and ignore ***Step 4***.

Step 4: Improve the accuracy of the approximate model during the simulation. To obtain higher accuracy, more optimal sampling points are needed using the Sequential Optimal Sampling (SOS) method. Go to ***Step 2***.

Step 5: Stop.

Additionally , according to the time-step t_0, the total number of iterations $N_{iteration}$ is obtained from following equation:

$$N_{iteration} = \frac{T_{simulation\_time}}{t_0} \tag{29}$$

where $T_{simulation\_time}$ is the simulation time of the original model.

4.2 Dealing with state variables

If a simulation model has the property of state variables, the model can be expressed as:

$$\begin{cases} f(\boldsymbol{u},\boldsymbol{s},\dot{\boldsymbol{s}},t)=0 \\ \boldsymbol{y}=g(\boldsymbol{s},\dot{\boldsymbol{s}},\boldsymbol{u},t) \end{cases} \tag{30}$$

where t is time variable, and u, s and y are the input, state and output variables, respectively.

Because of the integrated characteristic of the state variables, they need to be handled individually in the simulation. As shown in Figure. 7, when replacing the current module with the ISM model or approximate model during the simulation process, all of the M state variables (S_1, S_2, …, S_M) should be handled and self-loops are formed. We can see that M inputs and M outputs are added to the model, and the total number of inputs and outputs are P+M and K+M. The inputs S_1, S_2,…, S_M are only one time-step delay from the outputs S'_1, S'_2, …, S'_M, and the initial values of the inputs are set with S^0_1, S^0_2, …, S^0_M. The self-loop guarantees that all of the state variables can be updated during the simulation process.

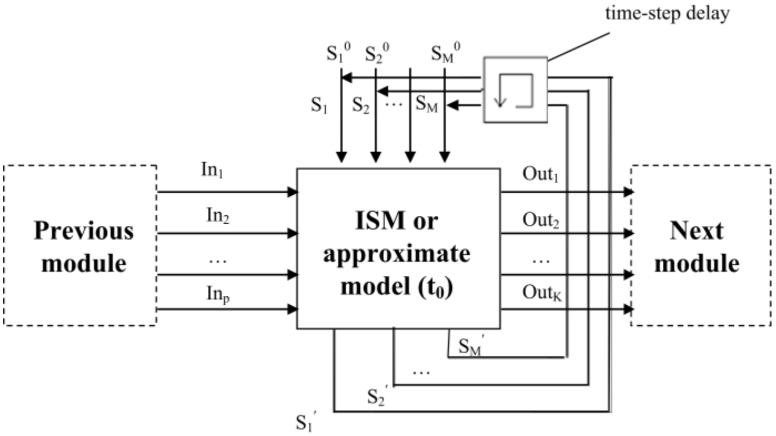


Figure 7: Handling the state variables in the procedure of simulation.

4.3 Performance metrics

The accuracy of the ISM model or the approximate model during the simulation process can also be measured by R^2 through off-design testing.

$$R^2 = 1 - \frac{\sum_{k=1}^{n} \left(y(x^k) - y_{approximate}(x^k)\right)^2}{\sum_{k=1}^{n} \left(y(x^k) - \overline{y_{approximate}}\right)^2} \tag{31}$$

where $y(x^k)$ obtained from the simulation model represents the actual value at the test point x^k, $y_{approximate}(x^k)$ is the value estimated by the approximate model at the test point x^k, $\overline{y_{approximate}}$ is the mean values calculated from the approximate model at all test points. The value of R^2 is between 0 and 1 and reveals the accuracy of the model. Basically, the closer to 1, the more accurate the tested model will be.

4.4 A simple test

The battery model was extracted from the pure electric vehicle model and was the equivalent circuit model of PNGV proposed by the US Department of Energy. It provided the voltage and current to drive the external load.

The battery model, as shown in Figure. 8, was tested under the condition of discharging. To realize discharging under the condition of constant power, a current module that can adjust current using PID control automatically was connected to the battery model.

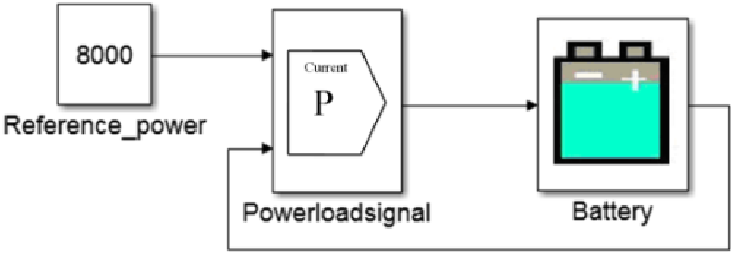


Figure 8: Diagram of the test simulation model.

The goal of this application was to approximate the battery model and replace the battery model in the original simulation model, as shown in Figure. 9. The ISM of battery was established as follows: two inputs and one output from the original component: V$_{in}$– the input voltage, P$_{external\_load}$– the reference power, V$_{out}$–the output voltage; one input and one output extracted from the state variable: SOC- the current state of charge, SOC$_{next}$-the state of charge of next step; the scope of input variables, V$_{in}$, P$_{external\_load}$, and SOC, were set according to the original simulation history data; time-step t_0 was set to 0.1 seconds, as shown in Table. 5. Besides, simulation time $T_{simulation\_time}$ and R^2 were set to 200 and 0.95. Then the final ISM model was established as shown in Figure. 10. Note that the variable values at red circle in Figure. 9 were needed to be saved for the inputs of ISM model. The parameters in Table. 5 were used to construct the approximate model.

The approximate model was constructed based on the ISM model using the RBF-based RSS method. Note that the obtained approximate model is in the form of RSS, it is needed to integrate the RSS to Simulink environment, which will be discussed in Section 5.1. Then the accuracy of the approximate model in Simulink environment needs to be verified. The battery of original simulation model was replaced with approximate model, and tests were carried out to verify the accuracy of approximate model, as shown in Figure. 11. The detailed settings of the tests will described in Section 5.1.

The comparisons between the simulation model with approximate component and the original simulation model and are shown in Figure. 12, and the performances about R^2 were calculated in Table. 6. As observed from Figure. 12, the output values, SOC$_{next}$ and V$_{out}$, of the simulation model with approximate component were nearly the same as the original simulation model, which meant that the approximate model had high precision.

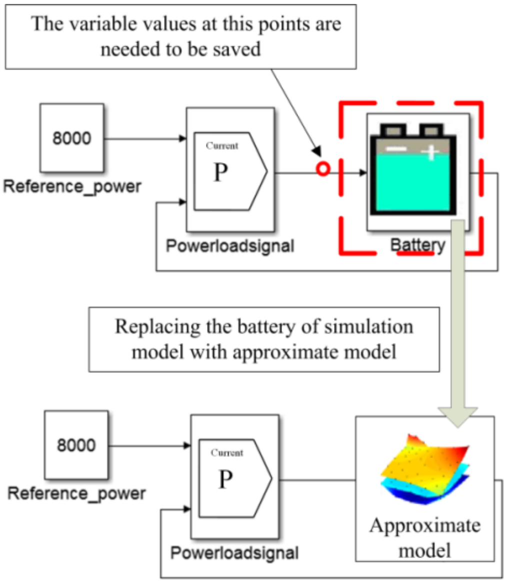

The variable values at this points are needed to be saved
8000
Reference_power
Current
P
Powerloadsignal
Battery
Replacing the battery of simulation model with approximate model
8000
Reference_power
Current
P
Powerloadsignal
Approximate model

Figure 9: The purpose of using approximate model.

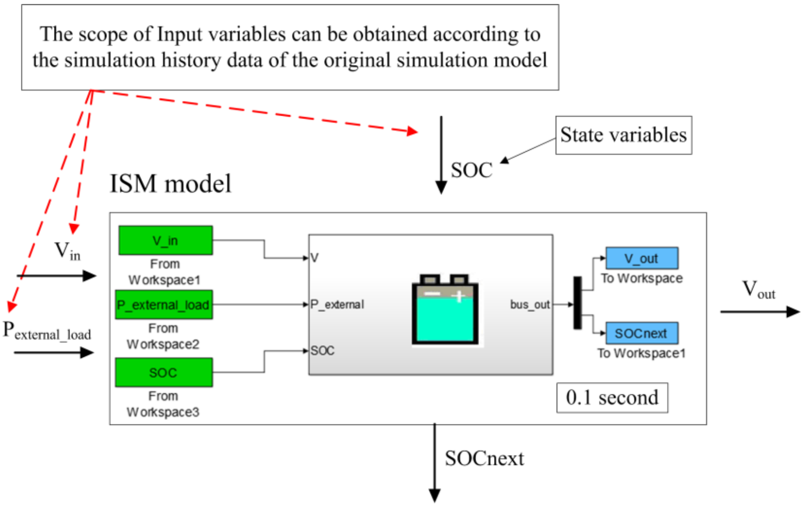


Figure 10: Constructing the approximate model using the ISM model.

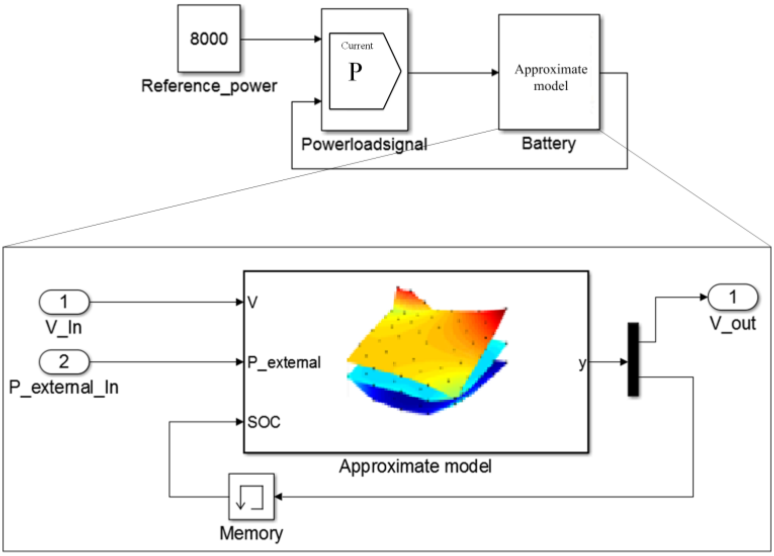


Figure 11: Verifying the accuracy of approximate component in simulation model.

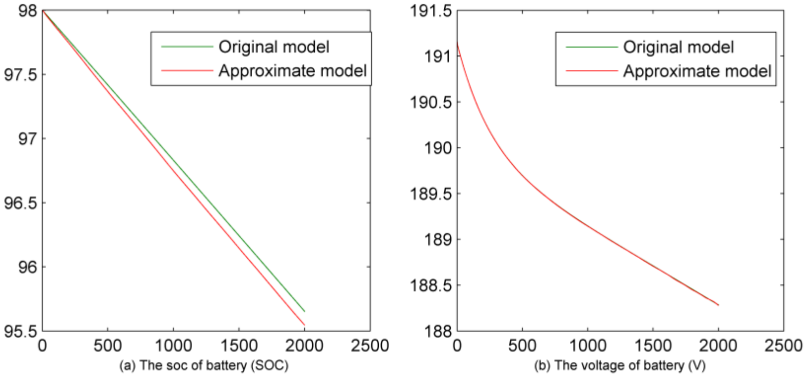


Figure 12: Comparisons between the simulation model with approximate component and the original simulation model.

Table 5: Parameters for constructing the approximate model.

Test cases	Inputs	Scope of variables	Outputs	D	q	M1	M2	δ
Battery	Vin	[188,191.2]	V_{out} SOC_{next}	2	3	1250	20	[1%,1%]
	Pexternal load	8000						
	SOC	[95.5, 98]						

Table 6: Performances results.

Performance metrics	SOC	V
R^2	0.9769	0.9899

5 An engineering application

As a part of the important government project “Research and application on key technologies of integrated development for pure electric vehicles”, “Multi-domain modeling and simulation for pure electric vehicle (PEV)” aims at building an entire PEV simulation model and doing some experiments to analyze the performances of the vehicle. The PEV simulation model is a complex system, including a motor, chassis, controller, battery, driver, etc., and is shown in Figure. 13.

The pure electric vehicle simulation model is built based on the experiments of the actual PEV. The layout of the PEV model is shown in Figure. 14, and PEV model is composed of driver, controller, powetrain module. The driver module provides drive cycle, namely the speed of the vehicle. The controller module calculates the power requirements of motor according to the drive cycle and the feedback signals of powertrain.

In this study, the driver followed the New European Drive Cycle (NEDC) to drive the vehicle. The NEDC is a driving cycle consisting of four repeated European urban (ECE-15) driving cycles and an extra-urban driving cycle (EUDC) as shown in Figure. 15. To explain the problem in a simple and clear way, the European urban driving cycle was used in this paper. Additionally, the road surface and wind resistance were considered in the simulation process.

To test the controller of a PEV, real-time simulation or HIL is necessary. Simplifying or approximating the powertrain is the important measure to realize real-time simulation, because usually the calculation of the original model of powertrain cannot meet the requirement of real-time simulation.

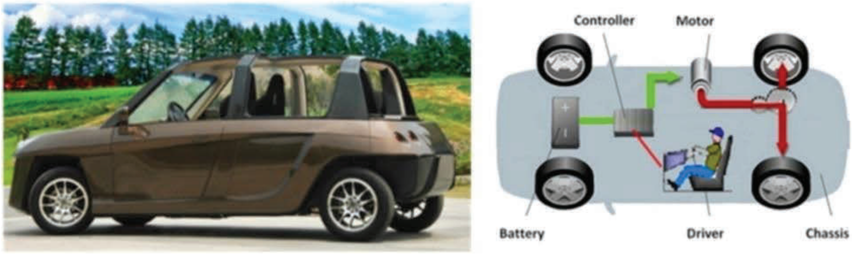


Figure 13: Pure electric vehicle: The appearance of the PEV (left) and the structure of the PEV (right)

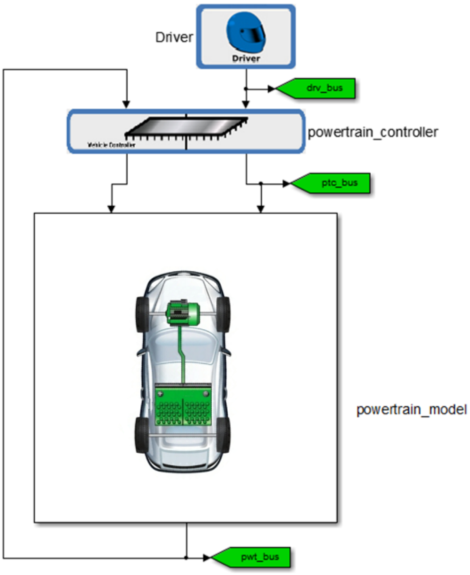


Figure 14: The pure electric vehicle simulation model.

5.1 Approximating the powertrain

The purpose of this application was to approximate the powertrain model and replace the powertrain model in the original simulation model, as shown in Figure. 16.

(1) Establish the initial ISM model

Before constructing the approximate model, the ISM model should be establised. The ISM model can be established as follows: two inputs, Cmd_acc – acceleration

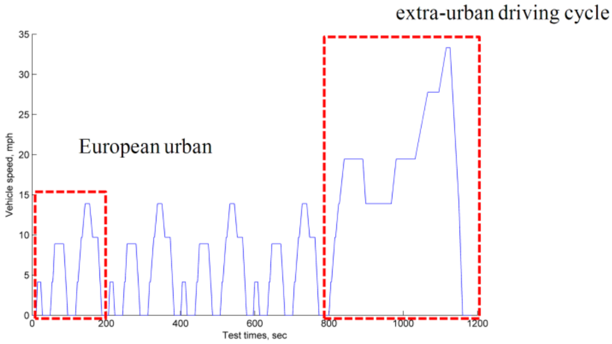


Figure 15: The NEDC drive-cycle.

command, Cmd_brake – break command; two outputs: W – the angular velocity of motor, Tau-the output torque of motor; two inputs and two outputs extracted from the state variables exist: V – the current vehicle speed, SOC–the current state of charge, V$_{next}$–the vehicle speed of next step, SOC$_{next}$–the state of charge of next step; the scope of input variables, Cmd_acc, Cmd_brake, SOC, and V, can be obtained according to the simulation history data of the simulation model; and the time-step t_0 was set to 0.1 seconds. Besides, simulation time $T_{simulation\_time}$ and R^2 were set to 200 and 0.95. Then the final ISM model is established as shown in Figure. 17, and the parameters in Table. 7 were used to construct the approximate model.

Table 7: Parameters for constructing the approximate model.

Test cases	Inputs	Space	Outputs	D	q	M1	M2	δ
PEV	SOC	[96.8,98]	Tau	2	3	6500	20	[1%,1%,1%,1%]
	V	[0,14]	W					
	Cmd_acc	[0,1]	V$_{next}$					
	Cmd_break	[0,1]	SOC$_{next}$					

(2) Construct the approximate model

Based on the ISM model, the approximate model can be expressed as follows:

$$\{Tau, W, V_{next}, SOC_{next}\} = RSS\_t_0\{Cmd\_acc, Cmd\_brake, V, SOC\} \tag{32}$$

The approximate model was constructed using the RBF-based RSS method. For the outputs, four corresponding response surfaces were generated with the same four input variables.

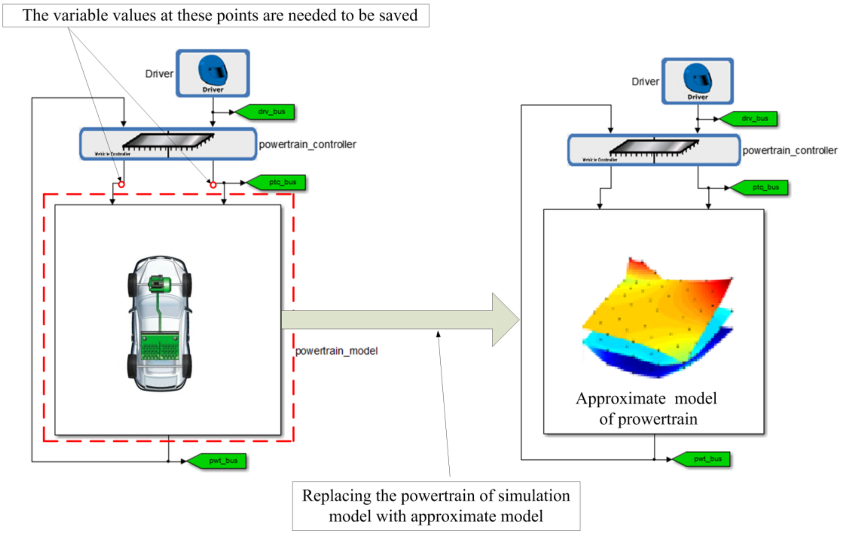

The variable values at these points are needed to be saved
Driver
drv_bus
powertrain_controller
ptc_bus
powertrain_model
pwt_bus
Approximate model of prowertrain
Replacing the powertrain of simulation model with approximate model

Figure 16: The purpose of using approximate model.

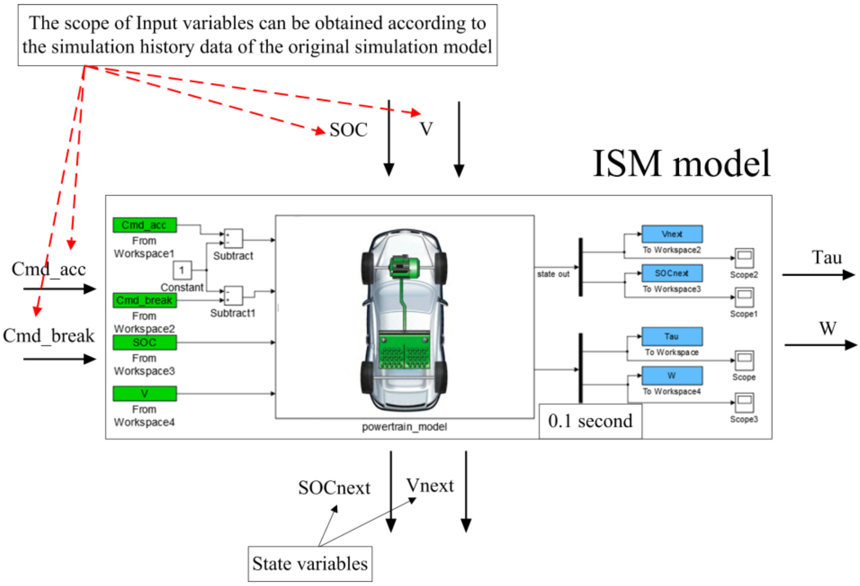


Figure 17: Constructing the approximate model using the ISM model.

(3) Test the approximate model

The obtained approximate model was in the form of RSS, and the powertrain of original simulation model in the Simulink environment cannot be replaced directly with RSS. The RSS has a reusable feature that can be used when needed. As the RSS is a mat-file, the block 'Embedded MATLAB function' can be used to integrate the RSS to Simulink environment. Then the powertrain of original simulation model was replaced with approximate component, as shown in Figure. 18.

The accuracy of approximate component in simulation model needs to be verified. The settings of the test were set as follows: the solver options in the configuration parameters of Simulink was set to fixed-step; the fixed-step size (fundamental sample time) was set to 0.1 seconds; the stop time was set to 200. The sample time of block was set to 0.1 seconds. The inputs and outputs of state variables were connected through the 'Memory' block.

The comparisons between the simulation model with approximate component and the original simulation model and are shown in Figure. 19, and the performances about R^2 of the approximate model are calculated in Table. 8.

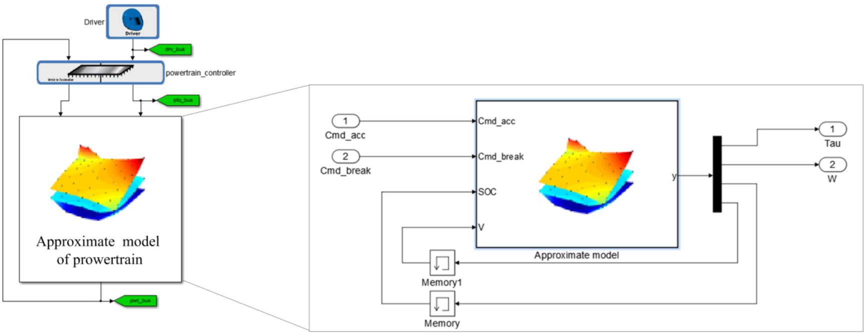


Figure 18: Verifying the accuracy of approximate component in PEV simulation model.

5.2 *Approximating results and analysis*

As observed from Figure. 19, the variable values of SOC, V, W, and Tau of the simulation model with approximate component are nearly the same as the values in the original simulation model.

However, the computational time between the original simulation model and the simulation model with approximate component is quite different. On a 2.70Ghz Pentium 4 PCs with 2 GB of random-access memory, it takes about 124 minutes

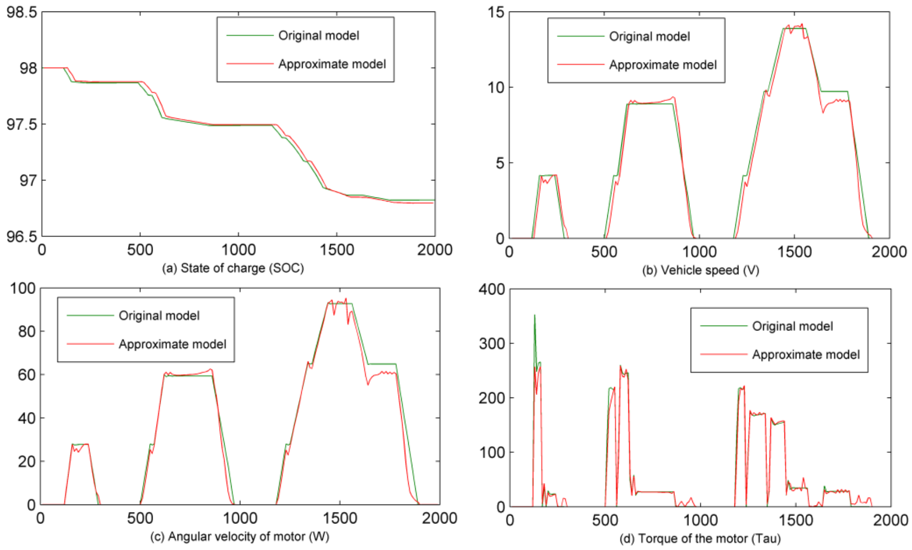


Figure 19: Comparisons between the simulation model with approximate component and the original simulation model.

to simulate the original simulation model. However, the time it takes to test the simulation model with approximate component is about 3 minutes. Apparently, the test process of the simulation model with approximate component lasts so short a time, as it is not needed to run the complex simulation model in each time-step t_0. Instead, the simulation data can be obtained directly from the approximate component in the sampling sites. Therefore, the efficiency of the simulation model with approximate component is much higher than the original simulation model. If the final approximate model meets the requirement of precision, the component in the original simulation model could be replaced by the approximate model, and the simulation model with approximate component can be used in real-time simulation or HIL.

Table 8: Performances results.

Performance metrics	SOC	V	W	Tau
R^2	0.9848	0.9745	0.9743	0.9811

5.3 Real-time simulation

To test the performance of the approximate model, a real-time test-bench was built based on the PCI extensions for Instrumentation (PXI) real-time platform. The PXI real-time platform is connected with a PC workstation by TCP/IP. Using TCP/IP, the codes (DLL) generated from the simulation model with approximate component through Real-Time Workshop (RTW) can be downloaded to the PXI real-time platform, and the simulation results in the PXI real-time platform can be sent back to PC workstation and shown on the display.

The PXI real-time platform mainly includes following hardware: PXI 1024Q box, PXI 8110RT controller, etc. The following softwares are required for the PC workstation: Matlab R2010b (including Simulink and RTW modules) LabView 2011sp1, LabView Real-Time 11.0, NI Verstand RT Engine 2011, Simulink Interface Toolkit (SIT), etc.

The process of importing the simulation model with approximate component in Simulink to LabView is illustrated in Figure. 20 The simulation model in Section 5.1 was used. To generate DLL for LabView to call, RTW was used which can translate the simulink model to DLL format. The SIT was used as an interface between Simulink and LabView, and user can download simulink model to PXI real-time platform using SIT server.

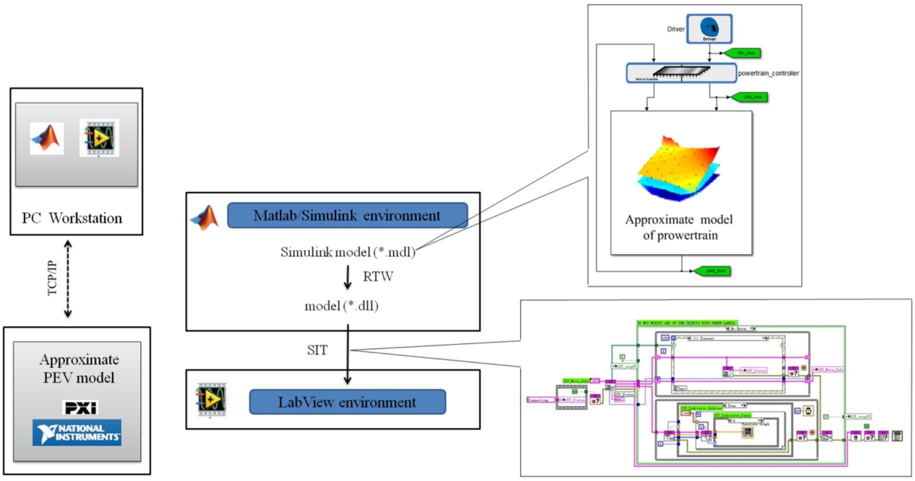
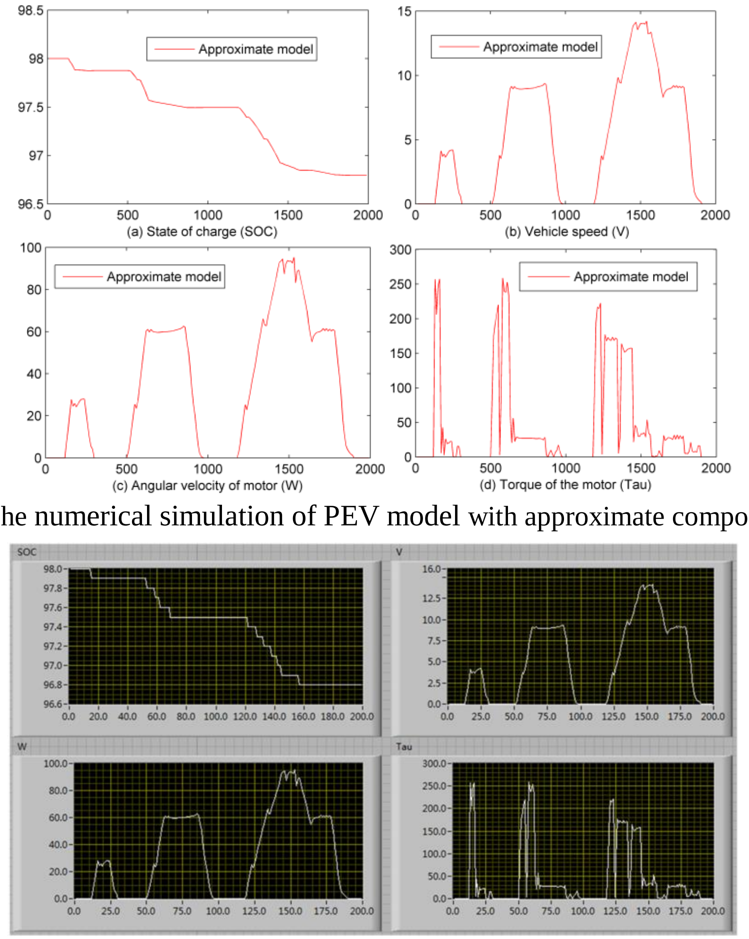


Figure 20: The process of importing the approximate model to LabView environment.

The fixed-step size of the test was set to 0.1 second as the requirements of real-time simulation, and the test results were presented in Figure. 21 As can be seen from these figures, the curves of the simulation model with approximate component obtained from real-time simulation were almost the same as the ones came from numerical simulation.

It is noted that we also tested the codes generated directly from original simulation model through RTW in the PXI real-time platform. However, we found that the PXI real-time platform failed to execute, and prompted the real-time performance of the tested model cannot meet the requirement (Fixed-step size: 0.1 second). We even set the fixed-step size to 10 seconds, the PEV simulation model still failed to run in the PXI real-time platform.



(a) The numerical simulation of PEV model with approximate component

(b) The real-time simulation of PEV model with approximate component

Figure 21: Comparison between numerical simulation and real-time simulation.

HIL simulation will be carried out in the next step. The controller of the PEV model will be replaced with real controller in the close-loop test system. We believed that the approximation method proposed in this paper can play an important role in HIL simulation.

6 Conclusions

This paper presented a novel global approximation method to approximate complex simulation models. Several mathematical and simulation model tests were carried out to demonstrate the feasibility and effectiveness of the proposed method. The proposed methods have some attractive features, as follows:

1. The complex MIMO simulation model can be approximated using the proposed method efficiently. After pretreatment, the complex simulation model with multiple inputs and multiple outputs can easily be approximated using the RBF-based response surface set method. The obtained approximate model can replace the complex simulation model, and can be used in many fields, such as parametric experiment, sensibility analysis, design or control optimization, etc.

2. The proposed method can address the dynamic features of the simulation model. The dynamic features, namely state variables, can also be addressed using the proposed method for the simulation model. The internal characteristics of the simulation model can be obtained from approximate model constructed using these state variables at any time.

3. The proposed method can be used in real-time simulation. As the complexity of simulation model, it is difficult to be solved in real-time steps. Therefore, an approximate model constructed using the proposed method can be used to realize real-time simulation.

However, as to digital simulation, it will be more efficient to change the time-step automatically. The proposed method needs to be improved to address the variable time-step.

Acknowledgement: This work is supported by the National 863 project (2013AA 041301), National Natural Science Foundation of China (No.51175198, No.512751 82, and No.51375186) and the Dongguan City special major science and technology projects (2011215155). We are also grateful to the anonymous referees for their insightful comments.

References

Beatson, R. K.; Newsam, G. N. (1992): Fast evaluation of radial basis functions: I. *Computers & Mathematics with Applications,* vol. 24, no.12, pp. 7-19.

Box, G. E. P.; Wilson, K. B. (1951): On the experimental attainment of optimum conditions. Journal of the Royal Statistical Society. *Series B (Methodological),* vol. 13, no.1, pp. 1-45.

Buhmann, M. D. (2000): Radial basis functions. *Acta Numerica 2000,* vol. 9, pp. 1-38.

Clarke, S. M.; Griebsch, J. H.; Simpson, T. W. (2005): Analysis of support vector regression for approximation of complex engineering analyses. *Journal of mechanical design,* vol. 127, no. 6, pp. 1077-1087.

Driscoll, T. A.; Heryudono, A. R. H. (2007): Adaptive residual subsampling methods for radial basis function interpolation and collocation problems. *Computers & Mathematics with Applications,* vol. 53, no.6, pp. 927-939.

Duncan, W. J. (1944): Some devices for the solution of large sets of simultaneous linear equations (with an appendix on the reciprocation of partitioned matrices). *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, Seventh Series,* vol. 35, pp. 660-670.

Elgohary, T. A.; Dong, L.; Junkins, J. L.; Atluri, S. N. (2014): Time domain inverse problems in nonlinear systems using collocation & radial basis functions. *CMES: Computer Modeling in Engineering & Sciences,* vol. 100, no. 1, pp. 59–84.

Elgohary, T. A.; Dong, L; Junkins, J. L.; Atluri, S. N. (2014): A Simple, Fast, and Accurate Time-Integrator for Nonlinear Dynamical Systems. *CMES: Computer Modeling in Engineering & Sciences,* vol. 100, no. 3, pp. 249 - 275.

Fang, H.; Horstemeyer, M. F. (2006): Global response approximation with radial basis functions. *Engineering Optimization,* vol. 38, no. 4, pp. 407-424.

Friedman, J. H. (1991): Multivariate adaptive regression splines. *The annals of statistics,* pp. 1-67.

Gu, J.; Li, G. Y.; Dong, Z. (2012): Hybrid and adaptive meta-model-based global optimization. *Engineering Optimization,* vol. 44, no.1, pp. 87-104.

Hardy, R. L. (1971): Multiquadric equations of topography and other irregular surfaces. *Journal of geophysical research,* vol. 76, no. 8, pp. 1905-1915.

Iske, A.; Levesley, J. (2005): Multilevel scattered data approximation by adaptive domain decomposition. *Numerical Algorithms,* vol. 39, no.1-3, pp. 187-198.

Jones, D. R. (2001): Direct global optimization algorithm. Encyclopedia of optimization. *Springer US,* pp. 431-440.

Liu, J. (2007): Modeling, configuration and control optimization of power-split hybrid vehicles. *The University of Michigan.*

Matheron, G. (1963): Principles of geostatistics. *Economic geology,* vol. 58, no.8, pp. 1246-1266.

Nguyen, X. S.; Sellier, A.; Duprat, F. (2009): Adaptive response surface method based on a double weighted regression technique. *Probabilistic Engineering Mechanics,* vol. 24, no. 2, pp. 135-143.

Park, J. W.; Venayagamoorthy, G. K.; Harley, R. G. (2005): MLP/RBF neural-networks-based online global model identification of synchronous generator. *Industrial Electronics, IEEE Transactions on 2005,* vol. 52, no. 6, pp. 1685-1695.

Peng, H.; Wu, J.; Inoussa, G. (2009): Nonlinear system modeling and predictive control using the RBF nets-based quasi-linear ARX model. *Control Engineering Practice,* vol. 17, no.1, pp. 59-66.

Peng, H.; Kitagawa, G.; Wu, J. (2011): Multivariable RBF-ARX model-based robust MPC approach and application to thermal power plant. *Applied Mathematical Modelling,* vol. 35, no. 7, pp. 3541-3551.

Prakasvudhisarn, C.; Trafalis, T. B.; Raman, S. (2003): Support vector regression for determination of minimum zone. *Journal of manufacturing science and engineering,* vol. 125, no. 4, pp. 736-739.

Roussos, G.; Baxter, B. J. C. (2005): Rapid evaluation of radial basis functions. *Journal of Computational and Applied Mathematics,* vol. 180, no.1, pp. 51-70.

Sacks, J.; Welch, W. J.; Mitchell, T. J. (1989): Design and analysis of computer experiments. *Statistical science,* pp. 409-423.

Simpson, J. R.; Landman, D.; Giroux, R. (2008): Adapting second-order response surface designs to specific needs. *Quality and Reliability Engineering International,* vol. 24, no. 3, pp. 331-349.

Vapnik, V.; Golowich, S. E.; Smola, A. (1997): Support vector method for function approximation, regression estimation, and signal processing. *Advances in neural information processing systems,* pp. 281-287.

Wang, G. G.; Shan, S. (2007): Review of metamodeling techniques in support of engineering design optimization. *Journal of Mechanical Design,* vol. 129, no. 4, pp. 370-380.

Wei, X.; Wu, Y. Z.; Chen, L. P. (2012): A new sequential optimal sampling method for radial basis functions. *Applied Mathematics and Computation,* vol.218, no.19, pp. 9635-9646.

Wu, J.; Peng, H.; Chen, Q. (2014): Modeling and control approach to a distinctive quadrotor helicopter. *ISA transactions,* vol. 53. no.1, pp. 173-185.

Zhen, H.; Qi, X.; Li, J. (2014): Neural Network Adaptive Control of MIMO Systems with Nonlinear Uncertainty. *The Scientific World Journal.*