




## Article

# Dimensionality Reduction, Modelling, and Optimization of Multivariate Problems Based on Machine Learning

Mohammed Alswaitti <sup>1,†</sup>, Kamran Siddique <sup>1,\*</sup>, Shulei Jiang <sup>1</sup>, Waleed Alomoush <sup>2</sup> and Ayat Alrosan <sup>2</sup>

<sup>1</sup> School of Computing and Data Science, Xiamen University Malaysia, Sepang 43900, Malaysia; alswaitti.mohammed@xmu.edu.my (M.A.); swe17091677@xmu.edu.my (S.J.)

<sup>2</sup> School of Information Technology, Skyline University College, Sharjah P.O. Box 1797, United Arab Emirates; waleed.alomoush@skylineuniversity.ac.ae (W.A.); ayat.alrosan@skylineuniversity.ac.ae (A.A.)

\* Correspondence: kamran.siddique@xmu.edu.my

† Current Address: SnT, University of Luxembourg, 6, Avenue de la Fonte, L-4364 Luxembourg, Luxembourg.

**Abstract:** Simulation-based optimization design is becoming increasingly important in engineering. However, carrying out multi-point, multi-variable, and multi-objective optimization work is faced with the “Curse of Dimensionality”, which is highly time-consuming and often limited by computational burdens as in aerodynamic optimization problems. In this paper, an active subspace dimensionality reduction method and the adaptive surrogate model were proposed to reduce such computational costs while keeping a high precision. In this method, the active subspace dimensionality reduction technique, three-layer radial basis neural network approach, and polynomial fitting process were presented. For the model evaluation, a NASA standard test function problem and RAE2822 airfoil drag reduction optimization were investigated in the experimental design problem. The efficacy of the method was proved by both the experimental examples in which the adaptive surrogate model in a dominant one-dimensional active subspace is given and the optimization efficiency was improved by two orders. Furthermore, the results show that the constructed surrogate model reduced dimensionality and alleviated the complexity of conventional multivariate surrogate modeling with high precision.

**Keywords:** active subspace; dimensionality reduction; surrogate model; optimization design; multivariate problems



**Citation:** Alswaitti, M.; Siddique, K.; Jiang, S.; Alomoush, W.; Alrosan, A. Dimensionality Reduction, Modelling, and Optimization of Multivariate Problems Based on Machine Learning. *Symmetry* **2022**, *14*, 1282. <https://doi.org/10.3390/sym14071282>

Academic Editor: José Carlos R. Alcantud

Received: 7 May 2022  
Accepted: 10 June 2022  
Published: 21 June 2022

**Publisher’s Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Optimal design usually involves multiple disciplines, multidimensional variables, and complex and time-consuming calculation models [1,2]. Thus, carrying out multi-point, multi-variable, and multi-objective optimization work would suffer the “Curse of Dimensionality” [3]. Combining big data analysis and machine learning, the intelligent optimization methods based on variable parameter dimensionality reduction and their application is the current development trend, which can reduce time and space complexity, save the overhead of unnecessary features, and improve the optimization effect [4–7].

In complex physical systems, scientists and engineers study the relationships between the models’ inputs and outputs. They employ computer models to estimate the parameters and their effects on the system. However, the process becomes intricate—if not impossible—when the simulation is expensive and the model has several inputs. To enable such studies, the engineers may attempt to reduce the dimension of the model’s input space.

Active subspaces are an emerging set of dimension reduction tools that identify important directions in the parameter space. Reducing the dimension can enable otherwise infeasible parameter studies [8]. Hence, the research contribution of this paper is based on active subspaces to explore and use the dimensionality reduction feature structure existing in the optimization problems. The internal main active features were extracted by using

the samples of input and output, transforming the high-dimensional optimization problem into a low-dimensional subspace to be handled. Artificial Neural Networks (ANNs) and other machine learning algorithms were then employed to build high-efficiency and high-precision surrogate models, which can both ensure the design precision and boost the optimization speed. The main contribution is to establish a lower-dimensional high-fidelity surrogate with ANN and polynomial fitting for efficient optimization design.

The rest of the paper is structured as follows: Section 2 presents related works and essential backgrounds. Section 3 presents the details of the proposed mechanism for constructing the surrogate model based on the radial basis function (RBF) with its pre and post-phases. The experimental design is presented in Section 4, and the obtained results and discussions are listed under Section 5. Finally, Section 6 includes the paper's conclusion.

## 2. Related Work

Surrogate models are used to perform simulations of complex systems. The cost of constructing accurate surrogate models with several input parameters increases exponentially, especially for time-consuming optimization problems. The dimensionality of the input sample will make the surface fitting process computationally difficult to achieve in some situations, leading to an efficiency bottleneck.

Dimensionality reduction of variables [9] and corresponding surrogate model installation in reduced dimensions need to be studied by transforming high-dimensional problems into low-dimensional problems. A simple dimensionality reduction method could be carried out using sensitivity analyses [10] to determine which design parameters have a larger influence on the system response. The less insensitive parameters are neglected to save the dimensionality considered in the surrogate regardless of some input variables that may lead to a low-fidelity model [11]. The low-fidelity surrogate model cannot satisfy the precision of nonlinear multivariate problems, e.g., the transonic airfoil [12].

Another dimensionality reduction strategy established by an effective surrogate model is to find an active subspace in a lower dimension in the entire variable space, which represents the maximum change directions of the system response and input variables. Compared with traditional dimensionality reduction methods, this approach can hold the maximum influential effects of design variables on the objective, thus providing an equivalent high-fidelity surrogate model of a much lower dimension.

The concept of the active subspace was introduced by Russi [13] and formalized by Constantine et al. [14]. Due to the efficiency of these dimensionality reduction techniques, active subspaces can be used and studied in some engineering and mathematical problems [4,8,15–17]. Recently, many studies have been proposed to tackle the real world and industrial problems such as the optimization of the industrial copper burdening system [18], the surrogate model for low Reynolds number airfoil based on transfer learning [19], and parameter reduction of composite load model proposed by the Western Electricity Coordinating Council [20]. Moreover, In the research proposed by Wang et al. [21], twenty-two dimensional functions related to airfoil manufacturing errors were approximated through the response surface in the one-dimensional active subspace. The work demonstrated that the uncertainty of the aerodynamic performance can be significantly reduced using a measurement information function that selects a small number of inspection points on the airfoil surface.

### 2.1. Active Subspace

Based on active subspaces, parameter dimensional reduction can take place to reduce the time and space complexity, save the overhead of unnecessary features, and improve the optimization effect. The mapping from input samples to output models can be seen as a multivariate function. Mostly, the engineering models contain several parameters. The active subspace consists of a low-dimensional subspace compared with the input sample space. The low-dimensional subspace represents the majority of the variability in the objective function. However, active subspaces seem to identify the low-dimensional

subspace compared with the space of inputs. In other words, active subspaces identify a set of important directions. Through these directions, simulation prediction changes fastest where other directions will not affect the prediction of the simulation so that it can be neglected.

From the dimensional analysis perspective, the result is that when the measurement units are changed, the relationships between physical quantities do not change. For example, the relationship between the speed at which an object falls to the ground and the height at which the object falls does not depend on whether the height is in feet or meters. Many different learning algorithms can achieve this relationship where each specific learning algorithm will generate a model with a specific inductive bias. The inductive bias of this learning algorithm is very important since different learning algorithms have different inductive biases. Consequently, with the same training data, different learning algorithms will generate different results. Therefore, the selection of the learning algorithm depends on the real problem to be solved.

There are some requirements to determine whether the simulation model is qualified to use an active subspace. The simulation model and the inputs should be well defined, where each input should have a range and enough resources to run the simulation model. These requirements include the following:

Normalized inputs: a vector  $x$  should include  $m$  components, each component  $\in [-1, 1]$ . The purpose of the normalization is to remove the possibility that some components may be too large, which can greatly affect the result, and some may be too small so that their effect could be ignored [8].  $x_\ell$  is a vector with  $m$  components which represents the lower bounds of vector  $x$ , and  $x_u$  represents the upper bounds of vector  $x$ . The computation method to normalize the input is as follows

$$\frac{1}{2}(\text{diag}(x_u - x_\ell)x + (x_u + x_\ell)). \quad (1)$$

Sampling density  $\rho$ : checking the dimension reduction sampling randomly is very useful in high dimensional space. Engineers must choose  $\rho$ , where different  $\rho$  values represent different results. Determining which  $\rho$  value is better depends on the suitability of each  $\rho$  to represent the parameter variability in each situation. There is no universal prescription for choosing  $\rho$  in the active subspace whereas the only constraint is that  $\rho$  must be a probability density. Active subspace mostly comes out from the gradient  $\nabla f(x)$  concept, and it is necessary to have the ability to evaluate the gradient. As in (2), the formula produces eigenvectors and eigenvalues. The eigenvectors are used to represent the active subspace dimension, and the eigenvalues that correspond to the eigenvectors are used to determine the active subspace [8]. First, it computes each  $x_i$  corresponding gradient  $\nabla f(x_i)$ . Then it calculates the eigenvalue decomposition.  $W$  represents the orthogonal matrix of eigenvectors, and  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$  represents the diagonal matrix of nonnegative eigenvalues.

$$C \approx \hat{C} = \frac{1}{M} \sum_{j=1}^M \nabla_x f(x_j) \nabla_x f(x_j)^T = \hat{W} \hat{\Lambda} \hat{W}^T \quad (2)$$

From the above, eigenvectors are used to represent the active subspace dimension, and the eigenvalues correspond to the eigenvectors, which are used to determine the active subspace. So, finding the gap in eigenvalues and choosing the first  $k$  eigenvalues to represent the active subspace is needed.

In complex systems, the active subspace works well provided that they have two features. The first is where the output changes monotonously with respect to the parameters when scientists informally describe the influence of parameters on the model. The second is to give the model a nominal parameter value in theoretical or actual measurement situations. Back-of-the-envelope estimates of the variation of the input parameters usually adopt a perturbation form of 5–10% relative to the nominal value. This perturbation

often attributes little change to the model. If the input perturbations are within the range, the linear model can represent the global trends well. These features are based on some complex models where many complex models do not have those features. Hence, using active subspace is not helpful in such systems [22].

In the quick-and-dirty check procedure, the least-squares method is required to come up with the coefficients of the linear regression model. The method of solving the model based on the minimization of the mean square error is called the least-squares method [23]. First, the data set  $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$ ,  $x_i = (x_{i1}; x_{i2}; \dots; x_{id})$ ,  $y_i \in \mathbb{R}$ , the sample has  $d$  attribute description. Then the coefficient of this equation could be calculated as follows

$$y_i \approx w^T x_i + b. \quad (3)$$

To find the most suitable coefficient,  $E_{\hat{W}} = (y - X\hat{W})^T (y - X\hat{W})$  should be the smallest where

$$X = \begin{bmatrix} 1 & x_1^T \\ \vdots & \vdots \\ 1 & x_m^T \end{bmatrix}, \hat{W} = (w; b), y = (y_1; y_2; \dots; y_m).$$

To minimize  $E_{\hat{W}}$ , let

$$\frac{\partial E_{\hat{W}}}{\partial \hat{W}} = 2X^T(X\hat{W} - y) = 0 \quad (4)$$

So that

$$\hat{w}^* = (X^T X)^{-1} X^T y \quad (5)$$

Then the suitable coefficient can be estimated using the least-squares method.

## 2.2. Surrogate Models

The surrogate models are very helpful for optimization problems that require multiple evaluations of the objective function. When each of the objective function evaluations takes a long time, the optimization itself becomes tricky. In particular, population-based evolution technology can use black-box simulators in parallel. However, in the presence of hardware constraints, there is a better method using a surrogate model to approximate the objective function first and then to use the surrogate model to perform the optimization process [24].

The surrogate model is considered as using the known independent and dependent variables' data to construct the mathematical relation expression. It is different from finding the unknown parameter in the mathematical relation expression. Therefore, the relational assumption can obtain an accurate surrogate model. The complete mathematical relation expression must be set up in the end where, generally, the mathematical relation expression is continuous. Although there are many discontinuous changes, the surrogate model is more suitable to analyze continuous change problems. The surrogate model is used to predict unknown sample points based on known sample points. Under different assumptions, the construction of the surrogate model can be different. Then the surrogate model can be divided into two types, namely, a regression model and an interpolation model.

The sample points usually contain a certain "noise" in the regression model so that this kind of surrogate model does not need to go through each sample point since in that way it cannot obtain the real mathematical relation. The regression model can not only predict the value of the unknown point but can also filter the noise of the sample data. In contrast, the interpolation model goes through each sample point and assumes each sample point to be correct. The value of the unknown point can be generated through the interpolation method. As the surrogate model is considered as an estimation of the known point to the unknown point, it has the error term. Hence, while constructing the surrogate model, the different assumptions and requirements of error can lead to different surrogate models [25].

The aerodynamic airfoil optimization process is very important in aircraft design. With the development of Computational Fluid Dynamics (CFD) techniques and the vast research on optimization algorithms in recent years, researchers have made great progress in aerodynamic shape optimization [25]. Using high-performance computers, the design period is especially shortened. However, useful parameterizations of airfoil or wing shapes for engineering models often result in high-dimensional design variables, so it is a great challenge to search for the optimum design due to the large solution space [9,26]. Aerodynamic designers now demand a quick and efficient approach to airfoil selection in burdensome engineering projects [10,11]. The key issue is to construct a high-fidelity efficient surrogate model that can obtain the desired solution as soon as possible [12].

There are some aspects to evaluate the performance of a surrogate model such as the generalization ability, the ability of the trained prediction model to correctly reflect samples that do not appear in the training set, ease of implementation, and the training speed. While training the surrogate model, both the prediction accuracy of the training set and the generalization ability are very important. However, these two features are not directly proportional. In some situations, such as for the regression surrogate model, when the prediction accuracy of the training set reaches a certain level, if it increases continuously, the accuracy of the sample point outside the training set will decrease. Therefore, the generalization ability will decrease.

A simple way to improve the generalization ability is to determine the parameters of the surrogate model many times to improve the prediction accuracy of the training set, compare it with the prediction accuracy of samples outside the training set, and eventually choose the better surrogate model between the trials. Another method is to cross-validate the sample points while training the agent model. Cross-validation means that when constructing a surrogate model, the sample points are divided into the model-building part and the verification prediction error part. For the same sample set, different divisions are performed to obtain different prediction errors and processed to obtain the prediction accuracy of the sample points in the sample set. Through cross-validation, more stable and reliable sample points can be obtained [25].

### 2.3. Artificial Neural Networks

One of the most popular surrogate models is the application of Artificial Neural Networks (ANNs) [27]. ANN methods became famous as universal function approximators because they provide good results on unseen data and their evaluation process is relatively computationally inexpensive. The back propagation (BP) neural network is the essence of the current ANNs. It is a network that contains multiple hidden layers that can deal with linear inseparable problems, but it is easy to fall into a local optimal solution and has a strong sensitivity to the initial weights. On the other hand, ANNs have strong learning capabilities, fault tolerance, large-scale parallel computing capabilities, fast computing speed, distributed storage capabilities, and very strong nonlinear mapping capabilities, etc. The radial basis function (RBF) method is a traditional method for multivariate function interpolation. Broomhead and Lowe [28] applied the RBF to the design of neural networks to construct RBF neural networks. It simulates a neural network structure in the human brain. For a certain local area of the network input space, only a few nodes will affect the output of the network, which is a local approximation network. Compared with other types of ANNs, RBF neural networks have a deep physiological basis, fast learning ability, excellent approximation performance, simple network structure, and have been widely used in many fields [29,30].

## 3. The Proposed Methodology

### 3.1. Finding an Active Space

To reduce the design variables' dimensionality and accelerate the optimizers, the methodology introduces the active subspace fundamentals. Active subspace can be used as an effective dimensional reduction strategy. It includes the positive semidefinite matrix

which consists of the average products of partial derivatives of the objective interest and the eigenvectors of a symmetric. If a lower dimension active subspace can be found, the active input samples' coordinates can be reduced to construct the equivalent surrogate model. The main feature of the active subspace is that it consists of important directions of the input space so that it has a lower dimension compared with the input dimension. The input samples create the largest change in these directions if the active subspaces exist in a multivariate problem since it is useful to find reduced coordinates. The average derivative of  $f$  is obtained as follows

$$C = \int (\nabla_x f)(\nabla_x f)^T \rho dx = W\Lambda W^T, \quad (6)$$

$$W = [W_1, W_2], \begin{bmatrix} \Lambda_1 & \\ & \Lambda_2 \end{bmatrix}, \quad (7)$$

where  $x \in [-1, 1]^n$  includes all the normalized input variables,  $\rho$  is the sampling density on the input space,  $\nabla_x f$  is the gradient of  $f$  with respect to  $x$ ,  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$  is the diagonal matrix of nonnegative eigenvalues, and  $W$  is the orthogonal matrix of eigenvectors. The eigenvalues measure how  $f$  changes on average [31] as follows

$$\lambda_i = \int \left( (\nabla_x f)^T w_i \right)^2 \rho dx. \quad (8)$$

Any  $x$  can be represented as  $x = W_1 y + W_2 z$  where  $y$  represents the active variables and  $z$  are the inactive variables. If  $\Lambda_1$  contains  $r < n$  comparatively larger eigenvalues, i.e., there exists a large gap between  $\lambda_r$  and  $\lambda_{r+1}$ ,  $W_1$  represents the first  $r$  eigenvectors and  $W_1^T x$  is the reduced coordinate  $y$ , i.e., the active variable [13]. Then, the approximation can be expressed as

$$f(x) \approx g(W_1^T x). \quad (9)$$

When the computational fluid dynamics (CFD) simulations for aerodynamic design have gradient capabilities (e.g., adjoint-based derivatives or algorithmic differentiation [32]), the formula in (6) and its eigenpairs can be obtained numerically using the Monte Carlo method or quadrature rules [8] as follows

$$C \approx \hat{C} = \frac{1}{M} \sum_{j=1}^M \nabla_x f(x_j) \nabla_x f(x_j)^T = \hat{W} \hat{\Lambda} \hat{W}^T. \quad (10)$$

For more common optimization designers that do not have subroutines for gradients, a diagnostic regression surface with least squares is employed for uncovering one-dimensional activity.  $M$  simulation samples are drawn independently based on the density  $\rho$  and

$$\hat{a} = \underset{a}{\text{argmin}} \frac{1}{2} \|\hat{u}a - f\|_2^2, f \approx \hat{u}a, \\ \hat{u} = \begin{bmatrix} 1 & x_1^T \\ \vdots & \vdots \\ 1 & x_M^T \end{bmatrix}, \hat{a} = \begin{bmatrix} \hat{a}_0 \\ \vdots \\ \hat{a}_n \end{bmatrix}, f = \begin{bmatrix} f_1 \\ \vdots \\ f_M \end{bmatrix}, \quad (11)$$

where  $\hat{a}$  is the coefficient describing the subspace combination structure. There are no eigenvalues computed in (12), but the vector  $a_{-0} = [\hat{a}_1, \dots, \hat{a}_n]^T$  actually approximates a one-dimensional active subspace linearly as

$$C \approx \int a_{-0} a_{-0}^T \rho dx = a_{-0} a_{-0}^T, \\ w_1 = a_{-0} / \|a_{-0}\|. \quad (12)$$

When a linear one-dimensional active subspace exists, this regression surface method with least squares is much more effective than the finite-difference in (10), which comes without huge computational cost and solver oscillation issues.

### 3.2. Equivalent Surrogate Construction

#### 3.2.1. The RBF Neural Network

The RBF neural network is a single hidden-layer forward neural network. The input layer is composed of some sensing units, which transfer the information to the hidden layer. A radial basis function is used as the activation function for the hidden layer, which has a larger number of neurons in the nonlinear conversion from the input space to the hidden layer space. The output layer acts as the linear combination of the hidden layer's output. As shown in Figure 1, there is an RBF neural network structure with  $I$  inputs,  $K$  hidden layer neurons, and  $M$  outputs.  $x_p = [x_{1p}, x_{2p}, \dots, x_{Ip}]^T$  is the  $p^{\text{th}}$  input sample, where  $p$  denotes the number of all input samples and  $I$  represents the input dimensionality.  $\varphi_p = [\varphi(x_1, c_k), \dots, \varphi(x_p, c_k)]^T \in \mathbb{R}^{p \times 1}$  is the output vector of the  $k^{\text{th}}$  hidden layer neuron corresponding to the  $p^{\text{th}}$  input sample, where  $c_k$  is the center of the neuron. Additionally,  $k = 1, \dots, K$ ,  $W = [w_1, w_2, \dots, w_M] \in \mathbb{R}^{(K+1) \times M}$  represents the output weight matrix, in which  $w_m = [w_{0m}, w_{1m}, \dots, w_{Km}]^T$  represents the connection weight between the hidden layer and the  $m^{\text{th}}$  output node.  $Y = [y_1, y_2, \dots, y_p]^T$  is the output matrix corresponding to the  $p$  input samples and  $\Sigma$  represents the linear activation function.

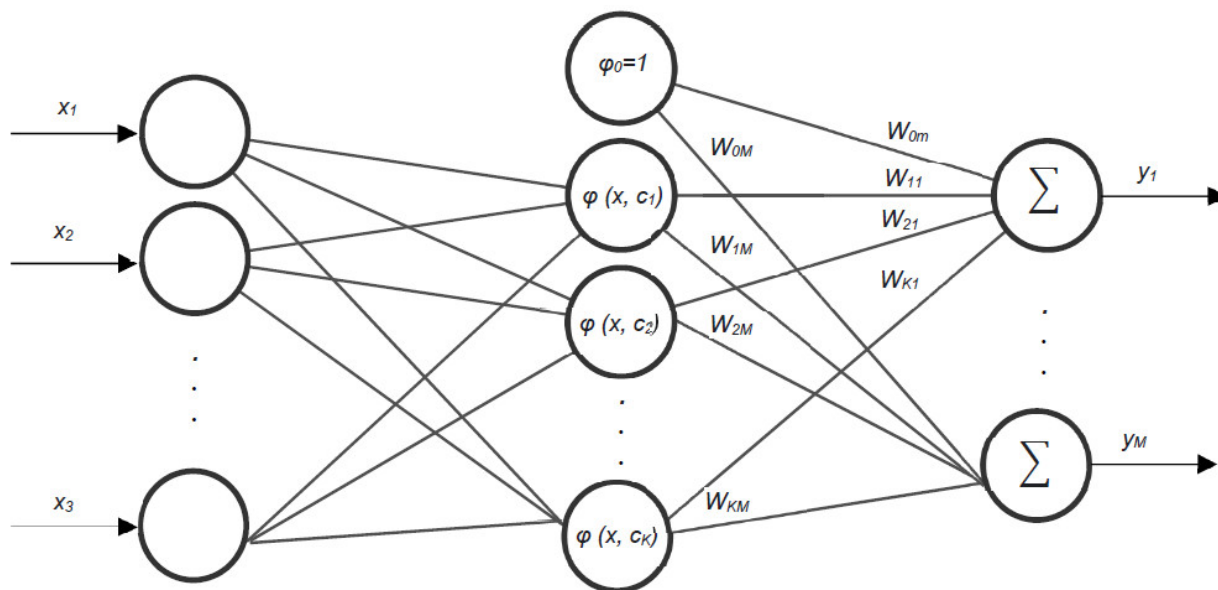


Figure 1. The RBF neural network structure.

The RBF network model is a flexible, reliable, and short-calculation-time surrogate model that can solve high-dimensional and high-order nonlinear problems well. The nature of the RBF network model is determined by the selected basis function. The Gaussian function is employed, which has the following formula:

$$\varphi(x, c_k, \sigma_k) = \exp\left(-\frac{\|x - c_k\|^2}{2\sigma_k^2}\right), k = 1, 2, \dots, K, \tag{13}$$

where  $\sigma_k$  is the width of the  $k^{\text{th}}$  hidden layer neuron.  $\|x - c_k\|$  is the geometry norm of  $x - c_k$ , denoting the radial distance of the sample  $x$  and  $c_k$ . Therefore, considering all the input samples, the  $m^{\text{th}}$  node output is calculated as follows

$$y_{pm} = w_{0m} + \sum_{k=1}^K w_{km} \varphi(x_p, c_k, \sigma_k), \quad (14)$$

and the actual output matrix of RBF is obtained as  $Y = \Phi W$ .

### 3.2.2. Polynomial Fitting

The least-squares method, which is the most common method to generate a polynomial equation from a given data set, is used. The concept of polynomial fitting is to use a polynomial expansion to fit all the input samples in a specified analysis area and determine the expansion coefficient. This method is useful in some simple models and easy to use. A polynomial function has the form as

$$y = a_0 + a_1x + \dots + a_kx^k, \quad (15)$$

where the residual is expressed as

$$R^2 \equiv \sum_{i=1}^n [y_i - (a_0 + a_1x_i + \dots + a_kx_i^k)]^2. \quad (16)$$

To find a suitable expansion coefficient that can minimize the residual, partial derivatives can be used. Making each result equal to zero for each direction, the expansion coefficient as in (17), leads to finding the best value to minimize the residual.

$$\frac{\partial(R^2)}{\partial a_k} = -2 \sum_{i=1}^n [y_i - (a_0 + a_1x_i + \dots + a_kx_i^k)] x_i^k = 0. \quad (17)$$

Then, by performing some matrix transformations [33], a simplified Vandermonde matrix can be obtained and expressed as  $X^T y = X^T X a$ . Finally, the expansion coefficients have an equation  $a = (X^T X)^{-1} X^T y$ .

## 4. Experimental Design

### 4.1. The Standard Test Function

For experimental design, the proposed method was tested with a standard test function problem and RAE2822 airfoil drag reduction optimization. The standard test function is analytical and is a well-known standard test problem in the NASA Langley MDO Test Suite with seven design variables [22]. The mathematical formulation is as follows

$$f(x) = 0.7854x_1x_2^2(3.3333x_3^2 + 14.9334x_3 - 43.0934) - 1.5079x_1(x_6^2 + x_7^2) + 7.477(x_6^3 + x_7^3) + 0.7854(x_4x_6^2 + x_5x_7^2) \quad (18)$$

where the variable bounds for the minimization problem are  $2.6 \leq x_1 \leq 3.6$ ,  $0.7 \leq x_2 \leq 0.8$ ,  $7.3 \leq x_4 \leq 8.3$ ,  $7.3 \leq x_5 \leq 8.3$ ,  $2.9 \leq x_6 \leq 3.9$ , and  $5.0 \leq x_7 \leq 5.5$ . Figure 2 shows what each of the  $x$  variables refers to.



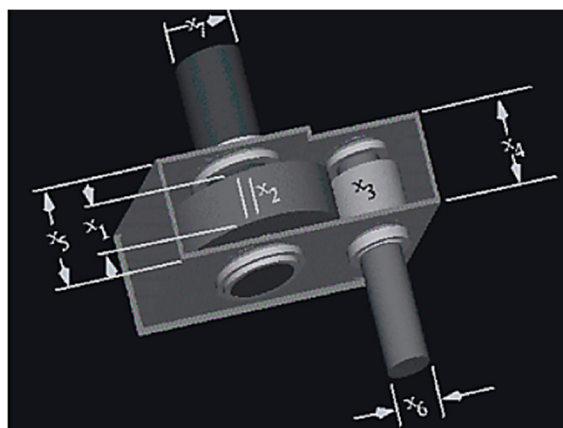


Figure 2. The diagram of the speed-reducer design problem.

4.1.1. Finding One-Dimensional Active Subspace

We can run a quick-and-dirty check to find whether this standard test function has a one-dimensional active subspace or not by the following steps:

- (1) N samples,  $\hat{x}_j \in [-1, 1]^m$ .
- (2)  $x_j = \frac{1}{2} [(x^u - x^l) \cdot \hat{x}_j + (x^u + x^l)]$ , where the dot operation is a component-wise multiplication.
- (3)  $f_j = f(x_j)$ .
- (4) Then, come up with the coefficients of the following linear regression model

$$f(x) \approx \hat{a}_0 + \hat{a}_1 \hat{x}_1 + \dots + \hat{a}_m \hat{x}_m \tag{19}$$

using least squares

$$\hat{a} = \underset{u}{\operatorname{argmin}} \|\hat{X}u - f\|_2^2, \tag{20}$$

where

$$\hat{X} = \begin{bmatrix} 1 & \hat{x}_1 \\ \vdots & \vdots \\ 1 & \hat{x}_N \end{bmatrix}, \hat{a} = \begin{bmatrix} \hat{a}_0 \\ \vdots \\ \hat{a}_m \end{bmatrix}, f = \begin{bmatrix} f_1 \\ \vdots \\ f_N \end{bmatrix} \tag{21}$$

- (5) Next, compute  $w = a' / \|a'\|$ , and  $a' = [\hat{a}_1, \dots, \hat{a}_m]^T$  which represents the coefficient of the linear regression approximation.
- (6) Lastly, plot the result where the first parameter is  $w^T \hat{x}_j$  and the second parameter is  $f_j$  [22].

After the dimensional reduction has been performed, the equivalent surrogate model construction is the next step. It starts by initializing m to 6, which represents the variables number, and N as the number of samples. Usually, N is initialized randomly within the range for each input sample variable  $x_1, x_2, x_4, x_5, x_6, x_7$ . All variables are represented by an  $N \times 6$  matrix that represents the total initialization value set

$$\begin{bmatrix} x_{u1}-x_{l1} & x_{u2}-x_{l2} & x_{u4}-x_{l4} & x_{u5}-x_{l5} & x_{u6}-x_{l6} & x_{u7}-x_{l7} \\ x_{u1}-x_{l1} & x_{u2}-x_{l2} & x_{u4}-x_{l4} & x_{u5}-x_{l5} & x_{u6}-x_{l6} & x_{u7}-x_{l7} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{u1}-x_{l1} & x_{u2}-x_{l2} & x_{u4}-x_{l4} & x_{u5}-x_{l5} & x_{u6}-x_{l6} & x_{u7}-x_{l7} \end{bmatrix}_{N \times 6},$$

where  $x_l$  represents the lower bound and  $x_u$  represents the upper bound of the sample value. After initialization, X is obtained as the value set after the normalization process where each element in X belongs to  $[-1, 1]$ . Then, the value of  $f(x)$  corresponding to

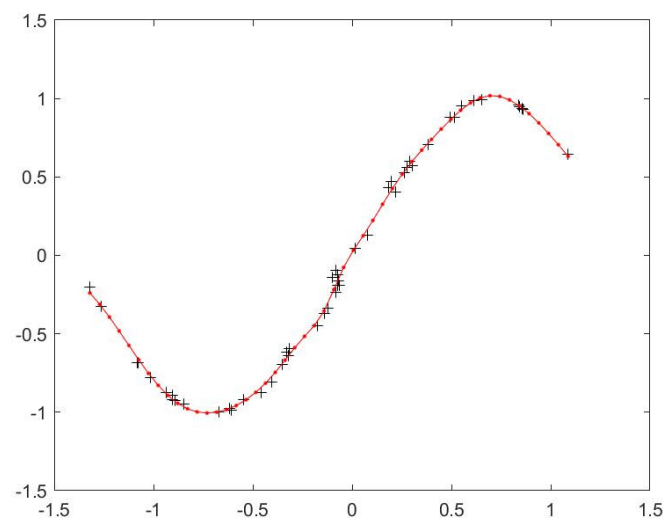
each input sample is calculated. The next step is to generate the coefficients of the linear regression model using the least-squares method, the basic technique that has already been discussed earlier. The coefficient equation is as follows

$$\hat{w}^* = (X^T X)^{-1} X^T y = (X)^{-1} (X^T)^{-1} X^T y = (X)^{-1} y \quad (22)$$

Lastly, plot the figure where  $x = X * w$ ,  $y = f$ . The result of  $X * w$  is an  $N \times 1$  matrix. Therefore, the input  $x$  which has six dimensions is converted to one dimension. If the figure shows the linear relationship between  $x$  and  $y$ , the one-dimension active subspace is found.

#### 4.1.2. The RBF Network Model

The RBF network model is a surrogate model that can solve high-dimensional and high-order nonlinear problems well. In general, two steps to are used to train an RBF network. The first is to determine the center of the neuron  $c_i$ . Next is to use the Back Propagation (BP) algorithm to determine the parameters  $w_i$  and  $\beta_i$ . It is worth mentioning that with this standard test function, it is not easy to use the RBF function. Hence,  $f(j) = \sin(x_1(j) + x_2(j))$  is employed in the dimensional reduction process and then the RBF is used to construct the surrogate model. From Figure 3, the black cross represents the input sample and the red line with points is the surrogate model.



**Figure 3.** The surrogate model using the RBF neural network of the standard test function.

#### 4.1.3. Performing a Polynomial Fit

After finding the active subspace, the next step is to construct the equivalent surrogate model. The  $Co = \text{polyfit}(yy, f, 5)$  function in Matlab is used to perform a polynomial fit. Both  $yy$  and  $f$  are the results of the one-dimension active subspace procedure.  $Co = \text{polyfit}(yy, f, 5)$  finds the coefficients for a polynomial  $f$  of degree five and it is the best fit for the data. The coefficients in  $Co$  are in descending powers, and the length of  $Co$  is equal to 6. After that, the obtained coefficients and the input  $yy$  are used to obtain the corresponding  $f$  using the polynomial function  $y = a_0 + a_1x + \dots + a_kx^k$ . Eventually, the surrogate model can be plotted as in Figure 3, which shows that the surrogate model achieved a perfect fit.

#### 4.1.4. Optimization Using the Genetic Algorithm

Lastly, the optimization using the genetic algorithm is performed. The goal of the standard test function is to find the maximum of  $f(x)$ , within the specified limited range. First, combine the dimensional reduction part with the polynomial fit part to be a function called a fitness function. This step is about performing the optimization based on the surrogate model. Compared with the process of optimization based on the original function,

using the surrogate model will reduce the computational cost. The fitness function can be used to tell the selection function, which sample  $x$  has a high possibility to remain, and which  $x$  needs to be abandoned. Through multiple iterations and selections, the final result  $x$  and the corresponding function value ( $fval$ ) can be generated using the genetic algorithm. The default global optimal solution for the genetic algorithm is to generate the global minimum while the standard test function aims to find the maximum of  $f(x)$ . Therefore, negating the objective function and finding the minimum value is equivalent to the maximum value of the original objective function.

#### 4.2. RAE2822 Airfoil Drag Reduction Optimization

RAE2822 airfoil drag reduction optimization is more complex, and a general optimization workflow was designed and is shown in Figure 4. A Latin hypercube sampling (LHS) [22] strategy based on the maximum criterion was employed to generate the training samples for identifying active subspaces and constructing an RBF neural network in a reduced dimension. To evaluate the aerodynamic performance, the baseline airfoil was fed as an input and then the class-shape transformation (CST) parameterization was used to obtain the parameter needed. After that, an NS solver was used to obtain a high-fidelity C-mesh. This experimental optimization formulation is as follows

$$\begin{cases} \text{find} & X^* \\ \text{min} & f = C_d \\ \text{s.t.} & |\Delta t_{\max}| \leq 0.05 * t_{0,\max} \\ & X \in [-0.01, 0.01]^{10} \end{cases} \quad (23)$$

where the optimization objective is to minimize the coefficient of drag  $C_d$  and the airfoil thickness of the maximum  $t_{\max}$  is taken as the constraint. For the design, in the optimization process, one/two-step optimization, i.e., gradient-based optimizers combined with the Multi-Island Genetic Algorithm (MIGA) [22], can be chosen and the constraints are handled by punishment [34]. The MIGA has the feature that each set of probable solutions is divided into several probable solutions called “islands”. On each island, the selection, crossover, and mutation operations can be performed separately. During this process, some of the individuals are selected from each island and some of the individuals are migrated to different islands in one iteration and roulette is used to carry out the selection. The advantage of the MIGA is that it can avoid the local maximum/minimum solution and suppress the chance of premature convergence. In the design, this work also considers a sequential least-squares quadratic program (SLSQP) algorithm for the optimum search. SLSQP has the property to solve nonlinear programming problems. The feasible optimum solution can be obtained efficiently by using the lower-dimensional RBF neural network and the two optimization algorithms as shown in Figure 5, which fully verifies the efficacy of the proposed method.

##### 4.2.1. Finding One-Dimensional Active Subspace and Polynomial Fitting

To evaluate the aerodynamic performance, the baseline airfoil is fed as an input and then the class-shape transformation (CST) parameterization is applied to obtain the required parameters. After that, using an NS solver, a high-fidelity C-mesh is obtained. Then, there are thirty groups of airfoil samples, each  $X$  has ten variables, and a corresponding drag coefficient  $f_m$ .

A function was designed to find the active subspace and the polynomial fit to construct the surrogate model. It is worth mentioning that using RBF only will not be sufficient and it will not provide a good fit. The process starts by loading the data shown in Table 1 and initializing  $m$  and  $N$  that represent the number of the original dimension and the number of the samples, respectively. Next is to initialize both the lower and upper bounds of  $X$ . Notably, the code is the same as the standard test function part where the only difference is the data.

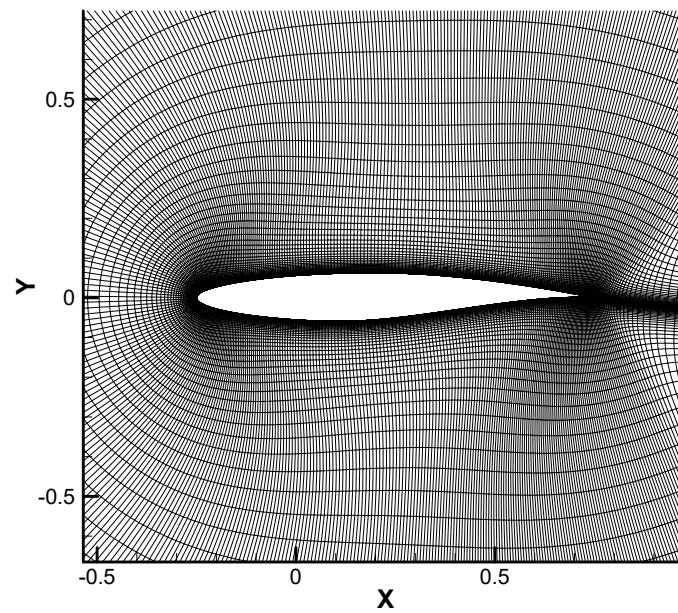


Figure 4. C-grid for RAE2822 airfoil optimization problem.

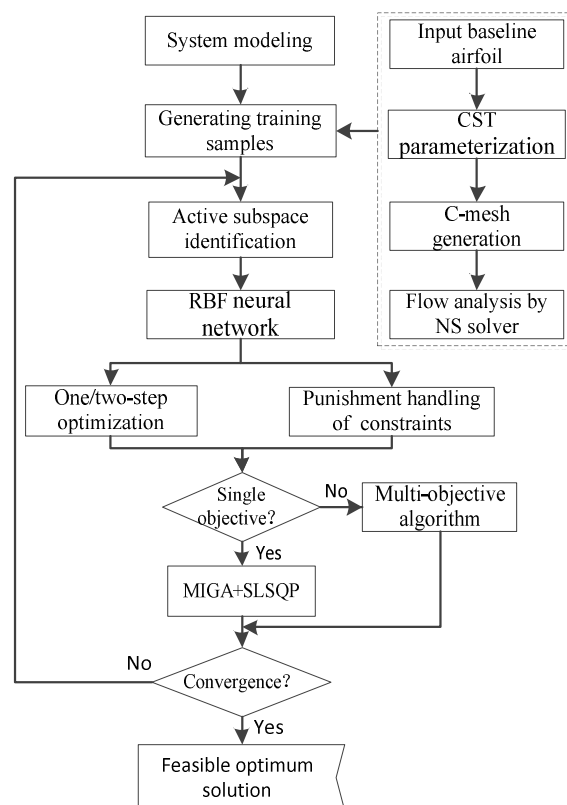


Figure 5. Implementation work-flow of the proposed method.

Table 1. Thirty Groups of Airfoil Samples.

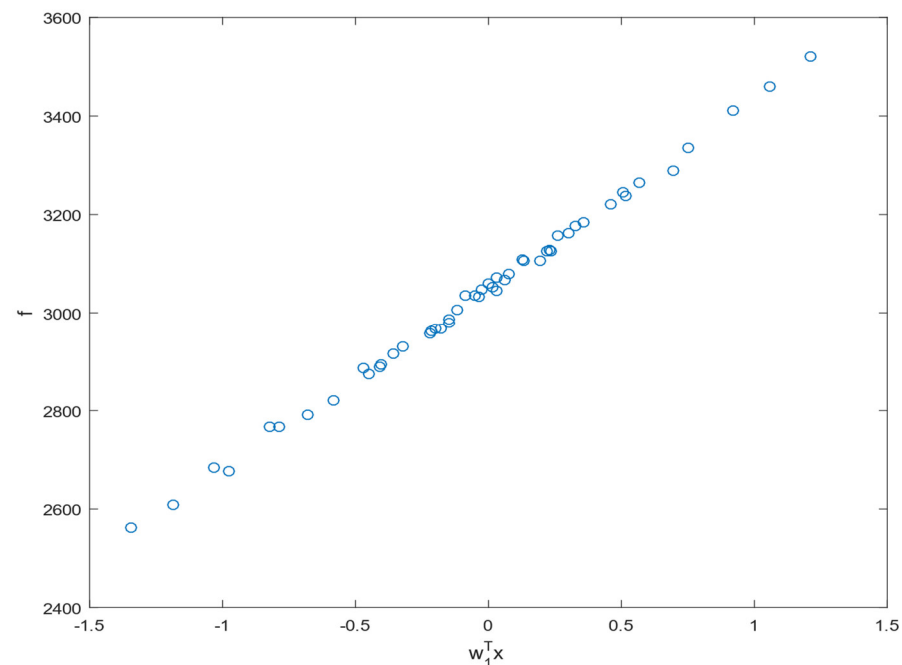
#	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	fm
1	5.2239E-03	8.3188E-03	−4.2653E-04	−7.8705E-03	−3.9396E-03	−8.7505E-03	6.0303E-03	3.7558E-03	4.7809E-04	−5.1587E-03	0.0104
2	7.5769E-04	3.0202E-03	−4.4409E-03	−8.2883E-03	8.7296E-03	4.3952E-03	−3.3393E-03	−9.4321E-04	6.1110E-03	−7.1830E-03	0.0115
3	5.5465E-03	−1.3677E-03	−4.6413E-03	−7.4978E-03	2.0037E-03	−2.5395E-03	−9.8637E-03	6.6072E-03	1.8095E-03	9.5083E-03	0.0099
4	−5.8693E-03	9.0404E-03	−3.7651E-03	4.4414E-03	−8.0366E-03	2.4585E-03	6.6284E-03	−1.7353E-03	−7.1610E-03	7.4779E-04	0.0122
5	−8.2223E-03	7.5726E-03	−2.0727E-03	3.1109E-03	1.3468E-03	−5.7459E-04	9.0071E-03	−6.3634E-03	−5.8676E-03	4.1196E-03	0.0124
6	8.1707E-04	7.9984E-03	−5.8060E-03	4.6342E-03	9.8523E-03	−7.9928E-03	−6.2891E-04	−9.6184E-03	−2.9229E-03	2.8257E-03	0.0108
7	5.0448E-03	−3.1049E-03	−6.3506E-03	6.8255E-03	−3.0960E-04	9.1091E-03	−5.0752E-03	3.0762E-03	−9.6682E-03	7.6554E-06	0.0122
8	−9.9445E-03	−5.4919E-03	7.6001E-03	5.9273E-03	−6.3772E-03	2.8449E-03	−3.2565E-03	−7.2955E-04	8.7269E-03	2.1741E-04	0.0112
9	−6.5070E-03	9.4273E-03	5.7492E-03	−9.0571E-03	−5.6415E-03	−5.9955E-04	4.2464E-04	3.1260E-03	−3.1011E-03	6.0278E-03	0.0110
10	8.6204E-03	−7.9403E-03	1.8081E-03	−9.7962E-03	5.8831E-03	−2.2466E-03	2.7385E-03	−4.7402E-03	7.3609E-03	−1.5914E-03	0.0111
11	−2.6034E-03	−5.2149E-03	5.0761E-03	7.4148E-03	8.3352E-03	8.5895E-04	−8.1410E-03	3.4674E-03	−6.3417E-03	−1.1206E-03	0.0110
12	5.5958E-03	−9.8474E-04	8.5413E-03	−5.9100E-03	2.4753E-03	1.8399E-03	−7.8765E-03	−3.9489E-03	6.7858E-03	−9.9150E-03	0.0109
13	−6.6130E-04	−7.8024E-03	−8.0007E-03	−2.9666E-03	7.7051E-03	3.3700E-03	5.6572E-03	−5.1858E-03	6.8478E-04	9.3021E-03	0.0124
14	−2.1074E-03	−6.0943E-03	7.8883E-03	9.8570E-03	−4.0314E-03	2.0526E-03	−1.4696E-03	4.4952E-03	1.9334E-03	−8.2352E-03	0.0112
15	7.4774E-03	8.2133E-03	−1.9441E-03	−3.9881E-03	−8.6936E-03	−6.5846E-03	1.6255E-03	2.5137E-03	−5.4360E-03	5.6222E-03	0.0104
16	−8.8679E-03	9.9088E-03	−7.3109E-03	−3.5621E-03	4.2742E-03	4.2929E-04	3.1330E-03	−6.9564E-04	7.8892E-03	−5.6389E-03	0.0114
17	−2.1208E-04	9.9879E-03	−4.5501E-03	6.1145E-03	4.6515E-03	−8.7713E-03	−7.7139E-03	−3.3496E-03	2.7838E-03	1.1211E-03	0.0098
18	−1.9226E-03	−9.6458E-03	−6.0210E-03	3.7168E-03	5.0344E-03	8.3876E-04	9.4140E-03	7.1223E-03	−4.7965E-03	−2.2746E-03	0.0122
19	8.0945E-03	−7.2009E-04	4.3260E-03	3.9173E-03	7.1523E-03	−8.1513E-03	−2.6276E-03	−7.8920E-03	−5.7090E-03	1.6529E-03	0.0105
20	6.9282E-03	1.4102E-03	4.1943E-03	−8.1628E-03	−7.5749E-03	−3.4882E-04	2.5600E-03	−4.4640E-03	−2.4143E-03	9.5677E-03	0.0113
21	4.6659E-05	9.2363E-03	2.7294E-03	−6.0931E-03	−8.4641E-03	−1.7497E-03	−2.9203E-03	5.9939E-03	7.0701E-03	−5.6481E-03	0.0103
22	−1.7438E-03	−8.6021E-03	7.5280E-03	−3.6632E-03	−4.8755E-03	−7.0142E-03	9.6291E-03	3.6970E-03	4.1864E-03	1.4500E-05	0.0109
23	3.2198E-03	−1.6536E-03	−3.9081E-03	6.2123E-03	−5.6820E-03	−9.3666E-03	−6.3436E-03	9.5249E-03	1.1314E-03	5.9499E-03	0.0093
24	−2.5373E-03	4.0397E-03	−8.6254E-03	8.8565E-03	−1.2233E-03	1.9692E-04	7.2079E-03	−5.3855E-03	3.5910E-03	−6.1257E-03	0.0120
25	−3.2448E-03	2.6240E-03	1.4064E-03	−7.1431E-03	−4.6324E-03	8.3810E-03	7.2690E-03	4.6024E-03	−1.2669E-03	−9.9292E-03	0.0127
26	5.5984E-03	−9.1964E-03	−2.1674E-03	−5.0888E-03	2.3077E-03	9.9186E-03	7.3413E-03	5.8130E-04	−1.7206E-03	−7.0018E-03	0.0131
27	−5.9895E-03	3.1427E-03	−6.5347E-03	9.0541E-03	−3.2169E-04	−8.3178E-03	1.4633E-03	6.4858E-03	−3.2183E-03	4.0661E-03	0.0103
28	8.4534E-03	6.8543E-03	−4.9004E-03	−2.5741E-03	4.9523E-04	−1.8872E-04	−6.2136E-03	2.5302E-03	−8.7092E-03	4.8387E-03	0.0108
29	−6.5977E-03	2.5726E-03	1.2790E-03	−1.4369E-03	6.0094E-03	5.8326E-03	9.0636E-03	−4.1876E-03	−9.4503E-03	−3.8241E-03	0.0133
30	3.8272E-03	−3.0399E-03	6.8947E-03	−5.3703E-03	5.4007E-03	−1.2038E-03	1.6030E-03	8.6865E-03	−9.3967E-03	−6.9194E-03	0.0112

#### 4.2.2. Optimization with Constraint

The goal of this optimal process is to use the genetic algorithm with a defined constraint  $|\Delta t_{\max}|_{0,\max}$ . Hence, in the genetic algorithm function, a constraint is added to satisfy this condition. The function is designed to obtain the maxthick from the function geometry with an inequality constraint. Furthermore, In the geometry function, the data of basic airfoil and data of thickness is loaded to generate the maxthick based on the input  $x$ . The first standard test function had only one objective function. However, in the airfoil, the thickness needs to be considered as the second objective. Under this constraint, finding the optimal result can be more complex and costly.

### 5. Results and Discussion

Firstly, the aim of the speed reducer design problem (as in (17)) is to minimize the weight of  $f(x)$ . Fifty training samples were used for active subspace identification. A scatter plot of  $w_1^T x$  versus  $f$  is given in Figure 6, which reveals a dominant one-dimensional active subspace.



**Figure 6.** A plot of the standard test function after one-dimensional reduction.

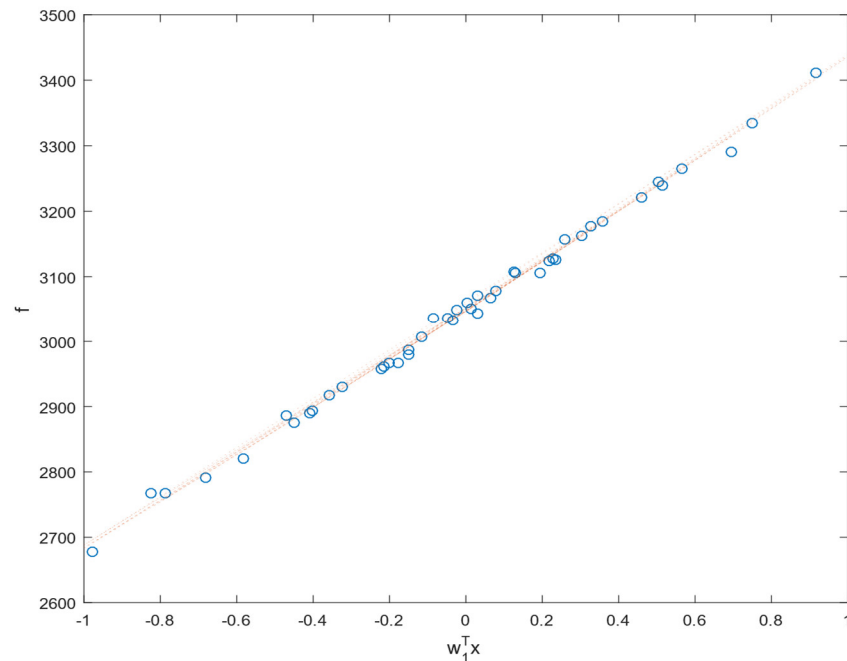
Figure 7 shows the blue hollow points which represent the input sample after dimensional reduction, and the red line represents the polynomial fitting that generates the surrogate model. This reveals the good fitting ability for polynomial fitting. In addition, it confirms the generation of a good surrogate model after fitting.

After optimization, the obtained results are shown in Table 2. Ten results for  $f(x)$  and the corresponding  $x_1, x_2, x_4, x_5, x_6, x_7$  values were recorded. It is concluded from the data that the minimum of  $f(x)$  was around 3915.55, and through calculating the variance of each variable  $x_i$ , the result shows that the variance was less than 0.05, which means that the optimization result is very stable.

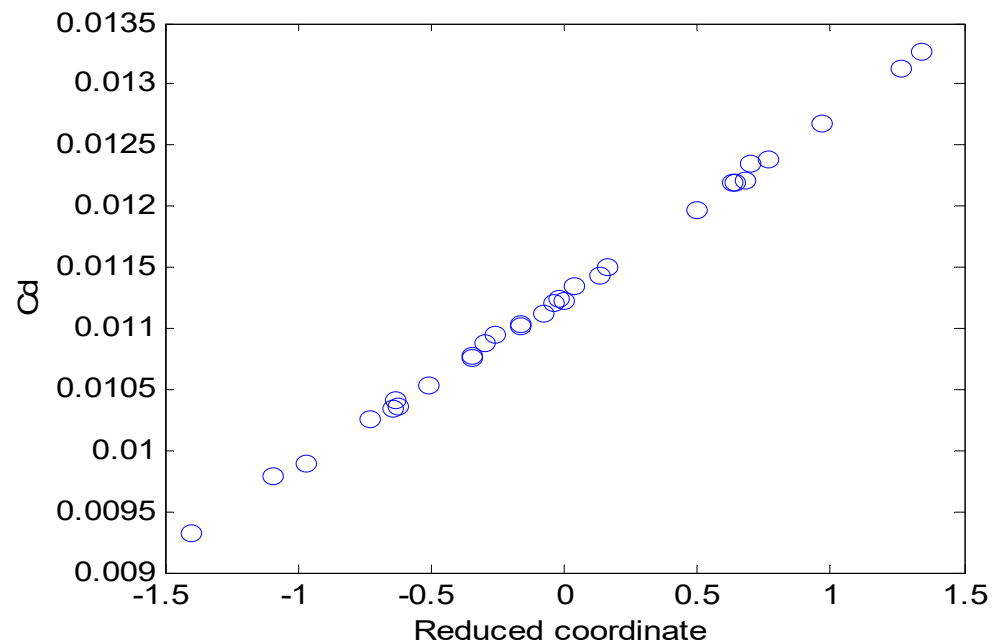
Secondly, RAE2822 airfoil optimization is considered the lift-constrained drag minimization. The optimization objective is to minimize the coefficient of drag  $C_d$ . The optimization is performed with conditions such as  $Ma = 0.729$ ,  $Re = 7.0 \times 10^7$ ,  $\alpha = 2^\circ$ . The optimization formulation is as in (23) where the variation of  $t_{\max}$  being within 5% is also taken as a constraint. In the experiment, 66,521 grids and 30 training samples were generated by a high-fidelity C-mesh of RAE2822 airfoil. These samples solved the problems of active subspace identification. A scatter plot of  $w_1^T x$  versus  $f$  is given in Figure 8, which

reveals a one-dimensional active subspace to enable a high-fidelity quadratic response surface model. The LOOCV was also employed for checking the approximation error, and the obtained result was a small MAE of  $1.14\text{E-}4$  as shown in Table 3.

Furthermore, the optimization results of the drag coefficient, thickness, and the corresponding variable  $x$  are shown in Table 4. The input baseline conforms to the standard RAE2822 airfoil whose CFD solution is shown in Figure 9.



**Figure 7.** A plot of the standard test function after polynomial fitting.



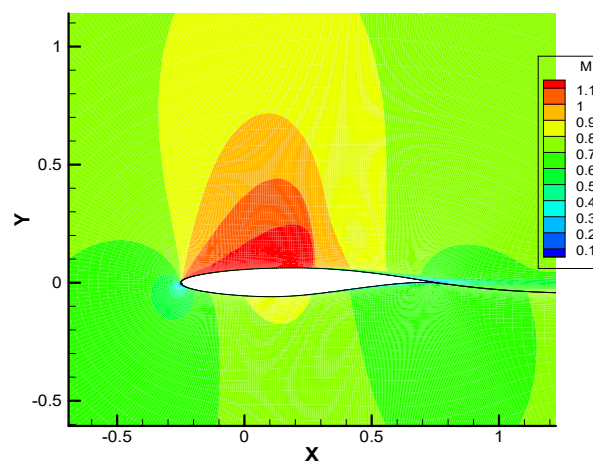
**Figure 8.** One-dimensional active subspace identified for RAE2822 airfoil.

**Table 2.** The optimization results for the standard test function.

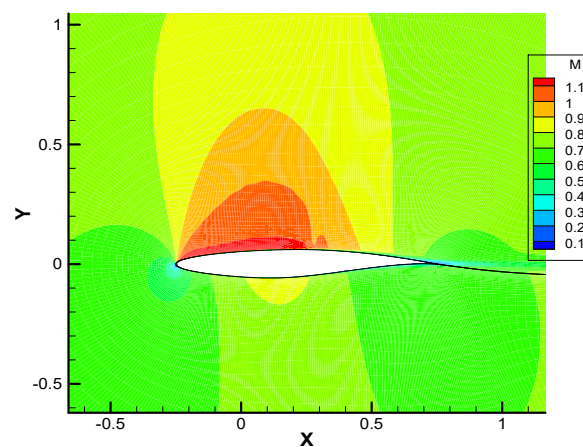
#	X1	X2	X4	X5	X6	X7	Result of F(x)
1	3.5957	0.7984	8.2682	8.2742	3.8993	5.4974	3.9067e+03
2	3.5969	0.7998	8.2727	7.7931	3.8983	5.4999	3.9213e+03
3	3.5997	0.7969	8.2968	8.2412	3.8973	5.4913	3.8957e+03
4	3.5995	0.7999	8.0032	8.2785	3.8980	5.5000	3.9264e+03
5	3.5986	0.7992	7.9973	8.2633	3.8918	5.4996	3.9357e+03
6	3.5865	0.8000	8.2437	8.2744	3.8999	5.4998	3.9262e+03
7	3.5991	0.7992	8.2812	8.2894	3.8945	5.4989	3.8955e+03
8	3.5995	0.7999	8.2813	8.2733	3.8999	5.4985	3.9311e+03
9	3.5890	0.7995	8.1505	8.2060	3.8940	5.4974	3.8973e+03
10	3.5960	0.7998	8.1445	8.2316	3.8972	5.4984	3.9196e+03

**Table 3.** Approximation and optimized result of RAE2822 airfoil.

	CFD Simulation	Error
MAE	-	1.14e−4
Optimized drag coefficient	0.0082	2.96%



(a)



(b)

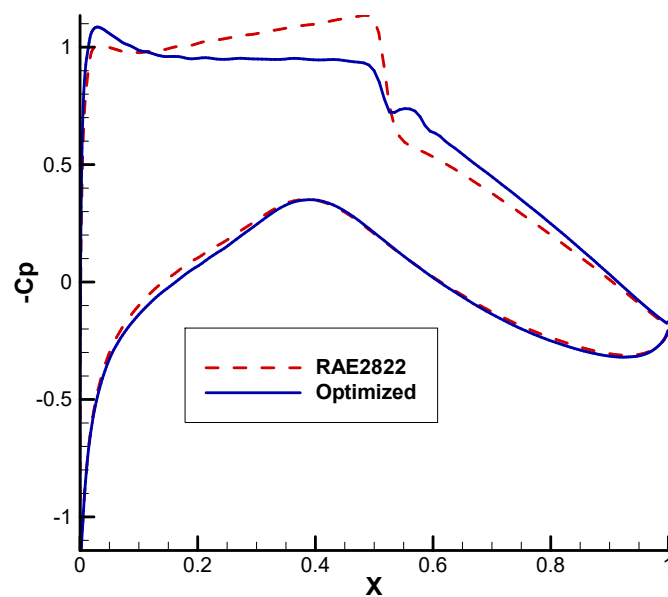
**Figure 9.** Comparison of the baseline and optimized geometries: (a) RAE2822 mach contours; (b) optimized mach contours.



**Table 4.** Optimization result of RAE2822airfoil drag problem.

#	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	fm	Thickness
1	0.0100	0.0100	0.0100	0.0100	−0.0100	−0.0100	−0.0100	0.0100	0.0100	0.0100	0.0082	0.1204
2	0.0100	0.0100	0.0100	0.0100	−0.0100	−0.0100	−0.0100	0.0100	0.0100	0.0100	0.0082	0.1204
3	0.0100	0.0100	0.0100	0.0100	−0.0100	−0.0100	−0.0100	0.0100	0.0100	0.0100	0.0082	0.1204
4	0.0100	0.0100	0.0100	0.0100	−0.0100	−0.0100	−0.0100	0.0100	0.0100	0.0100	0.0082	0.1204
5	0.0100	0.0100	0.0100	0.0100	−0.0100	−0.0100	−0.0100	0.0100	0.0100	0.0100	0.0082	0.1204

The convergence plot by the GA yields an optimized airfoil illustrated in Figure 9b. The optimality criterion was set as  $10^{-8}$ . Besides, the surface pressure coefficient distributions and shapes of the RAE2822 baseline and optimized airfoil are compared in Figures 10 and 11, respectively. Concerning the objective, the drag coefficient was optimized from 0.01123 to 0.0082 with approximately 27% reduction. Table 4 compares the CFD simulation and surrogate prediction of the optimized airfoil and verifies the design accuracy. As the design space  $[-0.01, 0.01]^{10}$  is not sufficiently large, the shock wave effects were not optimized to the minimum. However, the discovered active subspace improved the optimization efficiency by more than two orders compared with the traditional optimization using several hundreds of CFD simulations.

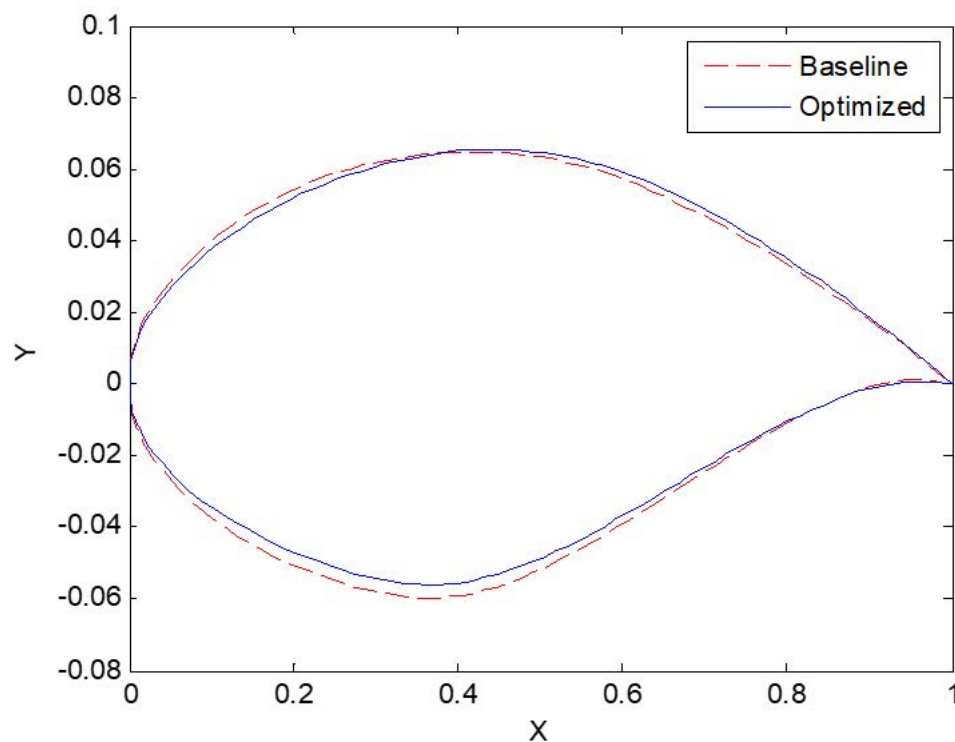
**Figure 10.** Pressure coefficient comparison of the RAE2822 baseline and optimized airfoil.

Lastly, the work aimed to find the existing active subspace and employ the reduced dimension sample to generate the surrogate model. The optimization result was obtained for the multivariate problem. In some other research works, a sensitivity analysis was traditionally employed [10] to determine which design parameters have a larger influence on the system response, and the less influential parameters are ignored. However, in the proposed method, all design variables were considered to generate a one-dimension active subspace, which has comparatively higher precision. In addition, other approaches usually combine the dimensional reduction with the polynomial fit. In this work, the RBF was formerly used to reduce the problem dimension and then to generate the surrogate model.

To further investigate the proposed method's performance, Table 5 lists a comparison between some of the latest methods and the proposed one. These results have been interpreted in the way they were reported in each work. The reported results prove that the proposed method has higher accuracy rates in optimizing the problem coefficients.

**Table 5.** Comparison of the aerodynamic coefficients for the RAE2822airfoil results.

Method	Cl	Error (Cl)	Cd	Error (Cd)
Zhang et al. [35]	0.683	8.07%	0.0138	8.69%
Wang et al. [36]	0.824	7.78%	0.0106	8.30%
Wu et al. [37]	0.824	8.20%	0.0113	7.79%
Present work	0.820	6.89%	0.0136	2.96%

**Figure 11.** Comparison of the baseline and optimized airfoil shapes.

The results show that both the standard test function and RAE2822 airfoil drag reduction optimization had a one-dimensional active subspace, which enabled an efficient surrogate model and accelerated the optimization design. The high precision of the second case was proved, and the result was improved after the optimization process. Therefore, combining big data analysis and machine learning, the intelligent optimization method based on the adaptive surrogate model was constructed in reduced dimensionality and its application is the current development trend. Hence, this proposed technique has great significance, which can reduce the time and space complexity, save the overhead of unnecessary features, and improve the optimization effect.

## 6. Conclusions and Recommendations

In this paper, the high effectiveness of the surrogate model usage in active subspaces was proved in both standard test function and RAE2822 airfoil drag reduction optimization. Compared with the traditional optimization using hundreds of time-consuming simulations, the optimization efficiency is expected to be improved by two orders, which would be greatly significant for existing real-world applications. The approximation error was rather small, which means that this process generates a high precision. The surrogate model constructed in reduced dimensionality also largely alleviates the complexity of conventional multivariate surrogate modelling. Future work can extend the advantages of the proposed method to more aerodynamic shape optimization or other related problems. Though a quick-and-dirty check for a one-dimensional active subspace has limitations, the average derivative of the  $f$  and the eigenvalues can be computed for  $r$ -dimensional active

subspaces, which would be widely applied. This method is of great significance to the current high-dimensional reduction field and also provides a beneficial reference for other methods in tackling such multivariate problems more effectively.

**Author Contributions:** Conceptualization and methodology, M.A.; validation and formal analysis, S.J.; investigation, M.A. and W.A.; data curation, S.J. and W.A.; writing—original draft preparation, A.A.; writing—review and editing, project administration, and funding acquisition, K.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by Xiamen University Malaysia Research Fund (Grant No: XMUMRF/2022-C9/IECE/0033).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Alomoush, W. Cuckoo Search Algorithm based Dynamic Parameter Adjustment Mechanism for Solving Global Optimization Problems. *Int. J. Appl. Eng. Res.* **2019**, *14*, 4434–4440.
- Alrosan, A.; Alomoush, W.; Norwawi, N.; Alswaitti, M.; Makhadmeh, S.N. An improved artificial bee colony algorithm based on mean best-guided approach for continuous optimization problems and real brain MRI images segmentation. *Neural Comput. Appl.* **2021**, *33*, 1671–1697. [[CrossRef](#)]
- Coello, C.A.C.; Brambila, S.G.; Gamboa, J.F.; Tapia, M.G.C.; Gómez, R.H. Evolutionary multiobjective optimization: Open research areas and some challenges lying ahead. *Complex Intell. Syst.* **2020**, *6*, 221–236. [[CrossRef](#)]
- Ma, L.; Huang, M.; Yang, S.; Wang, R.; Wang, X. An Adaptive Localized Decision Variable Analysis Approach to Large-Scale Multiobjective and Many-Objective Optimization. *IEEE Trans. Cybern.* **2021**, 1–13. [[CrossRef](#)] [[PubMed](#)]
- Alomoush, A.A.; Alsewari, A.A.; Alamri, H.S.; Zamli, K.Z.; Alomoush, W.; Younis, M.I. Modified Opposition Based Learning to Improve Harmony Search Variants Exploration. In Proceedings of the International Conference of Reliable Information and Communication Technology, Johor, Malaysia, 22 September 2019; pp. 279–287.
- Alomoush, A.A.; Alsewari, A.R.A.; Zamli, K.Z.; Alrosan, A.; Alomoush, W.; Alissa, K. Enhancing three variants of harmony search algorithm for continuous optimization problems. *Int. J. Electr. Comput. Eng. (IJECE)* **2021**, *11*, 2343–2349. [[CrossRef](#)]
- Alomoush, W.; Omar, K.; Alrosan, A.; Alomari, Y.M.; Albashish, D.; Almomani, A. Firefly photinus search algorithm. *J. King Saud Univ.-Comput. Inf. Sci.* **2020**, *32*, 599–607. [[CrossRef](#)]
- Constantine, P.G. *Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies*; SIAM: Philadelphia, PA, USA, 2015.
- Scott, D.W. The curse of dimensionality and dimension reduction. *Multivar. Density Estim. Theory Pract. Vis.* **2008**, *1*, 195–217.
- Yao, W.; Chen, X.; Luo, W.; Van Tooren, M.; Guo, J. Review of uncertainty-based multidisciplinary design optimization methods for aerospace vehicles. *Prog. Aerosp. Sci.* **2011**, *47*, 450–479. [[CrossRef](#)]
- Leifsson, L.; Koziel, S. Multi-fidelity design optimization of transonic airfoils using physics-based surrogate modeling and shape-preserving response prediction. *J. Comput. Sci.* **2010**, *1*, 98–106. [[CrossRef](#)]
- Li, W.; Krist, S.; Campbell, R. Transonic airfoil shape optimization in preliminary design environment. *J. Aircr.* **2006**, *43*, 639–651. [[CrossRef](#)]
- Russi, T.M. *Uncertainty Quantification with Experimental Data and Complex System Models*; University of California: Berkeley, CA, USA, 2010.
- Constantine, P.G.; Dow, E.; Wang, Q. Active subspace methods in theory and practice: Applications to kriging surfaces. *SIAM J. Sci. Comput.* **2014**, *36*, A1500–A1524. [[CrossRef](#)]
- Constantine, P.G.; Zaharatos, B.; Campanelli, M. Discovering an active subspace in a single-diode solar cell model. *Stat. Anal. Data Min. ASA Data Sci. J.* **2015**, *8*, 264–273. [[CrossRef](#)]
- Hu, X.; Zhao, Y.; Chen, X.; Lattarulo, V. Conceptual Moon imaging micro/nano-satellite design optimization under uncertainty. *Acta Astronaut.* **2018**, *148*, 22–31. [[CrossRef](#)]
- Hu, X.; Chen, X.; Parks, G.T.; Tong, F. Uncertainty-based design optimization approach based on cumulative distribution matching. *Struct. Multidiscip. Optim.* **2019**, *60*, 1571–1582. [[CrossRef](#)]
- Ma, L.; Li, N.; Guo, Y.; Wang, X.; Yang, S.; Huang, M.; Zhang, H. Learning to Optimize: Reference Vector Reinforcement Learning Adaption to Constrained Many-Objective Optimization of Industrial Copper Burdening System. *IEEE Trans. Cybern.* **2021**, 1–14. [[CrossRef](#)] [[PubMed](#)]
- Zhu, Z.; Guo, H. Design of an RBF Surrogate Model for Low Reynolds Number Airfoil Based on Transfer Learning. In Proceedings of the 2019 Chinese Control and Decision Conference (CCDC), Nanchang, China, 3–5 June 2019; pp. 4555–4559.
- Ma, Z.; Cui, B.; Wang, Z.; Zhao, D. Parameter Reduction of Composite Load Model Using Active Subspace Method. *IEEE Trans. Power Syst.* **2021**, *36*, 5441–5452. [[CrossRef](#)]

21. Wang, Q.; Chen, H.; Hu, R.; Constantine, P. Conditional sampling and experiment design for quantifying manufacturing error of transonic airfoil. In Proceedings of the 49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, Orlando, FL, USA, 4–7 January 2011; p. 658.
22. Constantine, P.G. A quick-and-dirty check for a one-dimensional active subspace. *arXiv* **2014**, arXiv:1402.3838.
23. Miller, S.J. *The Method of Least Squares*; Mathematics Department Brown University: Providence, RI, USA, 2006; Volume 8, pp. 1–7.
24. Hu, X.; Chen, X.; Parks, G.T.; Yao, W. Review of improved Monte Carlo methods in uncertainty-based design optimization for aerospace vehicles. *Prog. Aerosp. Sci.* **2016**, *86*, 20–27. [[CrossRef](#)]
25. Han, Z.-H.; Chen, J.; Zhang, K.-S.; Xu, Z.-M.; Zhu, Z.; Song, W.-P. Aerodynamic shape optimization of natural-laminar-flow wing using surrogate-based approach. *AIAA J.* **2018**, *56*, 2579–2593. [[CrossRef](#)]
26. Tang, G. *Methods for High Dimensional Uncertainty Quantification: Regularization, Sensitivity Analysis, and Derivative Enhancement*; Stanford University: Stanford, CA, USA, 2013.
27. Jun, T.; Gang, S.; Liqiang, G.; Xinyu, W. Application of a PCA-DBN-based surrogate model to robust aerodynamic design optimization. *Chin. J. Aeronaut.* **2020**, *33*, 1573–1588.
28. Broomhead, D.S.; Lowe, D. *Radial Basis Functions, Multi-Variable Functional Interpolation and Adaptive Networks*; Royal Signals and Radar Establishment Malvern: Malvern, UK, 1988.
29. Dash, C.S.K.; Behera, A.K.; Dehuri, S.; Cho, S.-B. Radial basis function neural networks: A topical state-of-the-art survey. *Open Comput. Sci.* **2016**, *6*, 33–63. [[CrossRef](#)]
30. Feruglio, L.; Corpino, S. Neural networks to increase the autonomy of interplanetary nanosatellite missions. *Robot. Auton. Syst.* **2017**, *93*, 52–60. [[CrossRef](#)]
31. Hu, X.; Zhou, Z.; Chen, X.; Parks, G.T. Chance-constrained optimization approach based on density matching and active subspaces. *AIAA J.* **2018**, *56*, 1158–1169. [[CrossRef](#)]
32. Hwang, J.T. *A Modular Approach to Large-Scale Design Optimization of Aerospace Systems*; University of Michigan: Ann Arbor, MI, USA, 2015.
33. Weisstein, E. Least Squares Fitting—from Wolfram Math World. [Online], [Retrieved on 4 March 2013]. Retrieved from the Internet at, Last Updated 2 March 2013. Available online: <https://mathworld.wolfram.com/LeastSquaresFitting.html> (accessed on 6 May 2022).
34. Zhou, Z.-H. *Machine Learning*, 1st ed.; Springer: Singapore, 2021.
35. Zhang, X.; Xie, F.; Ji, T.; Zhu, Z.; Zheng, Y. Multi-fidelity deep neural network surrogate model for aerodynamic shape optimization. *Comput. Methods Appl. Mech. Eng.* **2021**, *373*, 113485. [[CrossRef](#)]
36. Wang, Y.; Han, Z.-H.; Zhang, Y.; Song, W.-P. Efficient global optimization using multiple infill sampling criteria and surrogate models. In Proceedings of the 2018 AIAA Aerospace Sciences Meeting, Kissimmee, FL, USA, 8–12 January 2018; p. 0555.
37. Wu, X.; Zhang, W.; Peng, X.; Wang, Z. Benchmark aerodynamic shape optimization with the POD-based CST airfoil parametric method. *Aerosp. Sci. Technol.* **2019**, *84*, 632–640. [[CrossRef](#)]