Hindawi
Complexity
Volume 2018, Article ID 6950124, 19 pages
https://doi.org/10.1155/2018/6950124

Research Article

Evaluation for Sortie Generation Capacity of the Carrier Aircraft Based on the Variable Structure RBF Neural Network with the Fast Learning Rate

Tiantian Luan[1], **Mingxiao Sun**[1], **Guoqing Xia**[2] **and Daidai Chen**[3]

[1]*School of Automation, Harbin University of Science and Technology, Harbin, Heilongjiang 150080, China*
[2]*College of Automation, Harbin Engineering University, Harbin, Heilongjiang 150001, China*
[3]*College of Power and Energy Engineering, Harbin Engineering University, Harbin, Heilongjiang 150001, China*

Correspondence should be addressed to Mingxiao Sun; sunmingxiao@hrbust.edu.cn

Received 13 April 2018; Revised 15 August 2018; Accepted 17 September 2018; Published 22 October 2018

Academic Editor: Mahdi Jalili

The neural network has the advantages of self-learning, self-adaptation, and fault tolerance. It can establish a qualitative and quantitative evaluation model which is closer to human thought patterns. However, the structure and the convergence rate of the radial basis function (RBF) neural network need to be improved. This paper proposes a new variable structure radial basis function (VS-RBF) with a fast learning rate, in order to solve the problem of structural optimization design and parameter learning algorithm for the radial basis function neural network. The number of neurons in the hidden layer is adjusted by calculating the output information of neurons in the hidden layer and the multi-information between neurons in the hidden layer and output layer. This method effectively solves the problem that the RBF neural network structure is too large or too small. The convergence rate of the RBF neural network is improved by using the robust regression algorithm and the fast learning rate algorithm. At the same time, the convergence analysis of the VS-RBF neural network is given to ensure the stability of the RBF neural network. Compared with other self-organizing RBF neural networks (self-organizing RBF (SORBF) and rough RBF neural networks (RS-RBF)), VS-RBF has a more compact structure, faster dynamic response speed, and better generalization ability. The simulations of approximating a typical nonlinear function, identifying UCI datasets, and evaluating sortie generation capacity of an carrier aircraft show the effectiveness of VS-RBF.

1. Introduction

A carrier aircraft is an important part in the modern naval warfare. The research on the warfare capacity of the carrier aircraft has become a hot issue with the increasing attention of the security in the territorial sea. The comparison of sortie generation capacity of the carrier aircraft in different operational schemes is helpful to determine the final plan. Therefore, the evaluation for sortie generation capacity of the carrier aircraft has important theoretical significance and application value [1].

The evaluation for sortie generation capacity of the carrier aircraft is complex, due to the mutual influence and complex nonlinear of factors. The research of evaluation for the sortie generation capacity of the carrier aircraft has been studied recently. Xia et al. [2, 3] applied the principal component reduction method and the nonlinear fuzzy matter-element method to evaluate sortie generation capacity of the carrier aircraft. Both methods did not consider the mutual influence of factors. Gilchrist [4] proposed an evaluation method of the suitability of LCOM for modeling. This report studied the base-level munition production process in LCOM. However, the common evaluation methods ignore the correlation between the influencing factors. There is a certain deviation between the evaluation results and the actual situation.

The neural network has the advantages of self-learning, self-adaptation, and fault tolerance. It can establish a qualitative and quantitative evaluation model which is closer to human thought patterns. The trained neural network can

connect expert evaluation ideas with the neural network. Thus, this neural network can not only simulate the expert evaluation but also avoid the human errors in the evaluation process and the subjective influence of the human to calculate weight. The evaluation method based on the RBF neural network has advantages of fast calculation speed, high efficiency of problem solving, and strong self-learning ability. However, there are two problems in the application of the radial basis function (RBF) neural network.

The first problem is the structural design problem of the RBF neural network. In recent years, many optimization methods of the RBF neural network have been put forward.

(1) The pruning algorithm [5] was regarded as an effective way to optimize the network structure and improve the generalization ability of the network, but the parameters set in the pruning method required experience and skills

(2) The growing algorithm [6] increased the number of neurons and connections until the generalization ability met the requirements, but it was difficult to determine when to stop growing

(3) The pruning and growing algorithm: Kokkinos and Margaritis [7] present a Hierarchical Markovian Radial Basis Function Neural Network (HiMarkovRBFNN) model that enabled recursive operations. The hierarchical structure of this network was composed of recursively nested RBF neural networks with arbitrary levels of hierarchy. All hidden neurons in the hierarchy levels were composed of truly RBF neural networks with two weight matrices. The hidden RBF response units were recursive. However, this method was affected by the initial value, and sometimes, the final RBF neural network was unstable. However, the algorithm ignored the adjustment of structural parameters, which led to the slow convergence speed of the neural network learning algorithm. Therefore, the RBF neural network structure optimization design method is still an open problem, and especially the convergence of the dynamic structure adjustment process has not been solved well

The second problem is the training method for the weights of the RBF neural network and the learning rate. At present, RBF neural network weights are usually trained by the linear least square algorithm, but the least square estimation is affected by outliers [8, 9]. Kadalbajoo et al. [10] presented an RBF-based implicit explicit numerical method to solve the partial integro-differential equation which described the nature of the option price under the jump diffusion model. But the sum of squared errors increased rapidly with the increase of squared error of each training sample. At the same time, there is a problem to set the learning rate of the RBF neural network [11–14]. In the use of the RBF neural network, the learning rate was often subjective to set as a fixed value [15–20]. It remained unchanged throughout the learning process. If the learning rate was set too high, the convergence speed of the network might be very fast, and it might cause network instability. If the learning rate was too small, it would cause that the network convergence speed was slow and consume a large amount of computing time. Therefore, it is very difficult to choose a suitable learning rate for the traditional RBF neural network.

In order to solve the problems above, this paper proposes a variable structure RBF neural network (VS-RBF) with a fast learning rate. The number of neurons in the hidden layer is adjusted by calculating the output information (OI) of neurons in the hidden layer and the multi-information (MI) between neurons in the hidden layer and output layer. The robust regression method is used to replace the linear least square algorithm to reduce the influence of outliers on weight training. Then, the fast learning rate method is used to adjust the learning rate of the RBF neural network, which can guarantee the stable learning of the network and the convergence speed. In this paper, the proposed VS-RBF neural network can be used to grow or prune the neurons in the hidden layer according to the actual system.

The rest of this paper is organized as follows. Section 2 describes the VS-RBF neural network with a fast learning rate. Section 3 proves convergence of VS-RBF. In Section 4, VS-RBF is compared with SORBF and RS-RBF in approximating a typical nonlinear function and identifying UCI datasets. In Section 5, VS-RBF is used to evaluate sortie generation capacity of the carrier aircraft. Finally, conclusions are presented in Section 6.

2. Sortie Generation Capacity of Carrier Aircrafts

At present, the use of carrier aircrafts for domestic experience is very little. In theory, the index system of sortie generation capacity is established according to the foreign research results. The index of the sortie generation capability system established in this paper is based on the main factors pointed out in the 1997 “Nimitz” carrier aircraft exercise report. The factors are taken from the real environment and have high reference value. During the four-day exercise period, the equipment and environment did not affect the sorties. So the selection of indexes did not include the corresponding equipment and environment indexes [2, 3].

In this paper, when the index system of sortie generation capability is established, on one hand, we hope that the factors will be more comprehensive. Thus the credibility of the evaluation results will be increased. On the other hand, considering that if all the possible factors are added to the index system, the modeling difficulty will be greatly increased, and the evaluation system will also be proofread. So we should select the evaluation index according to the following principles:

(1) Considering the hierarchy and correlation between the evaluation indexes, if the high-level indexes can be obtained, the underlying indexes are not considered repeatedly. For example, one of the lowest levels is the carrier deck design, but the impact of the carrier deck design on the sortie generation capability is derived from the redeparture preparation time,

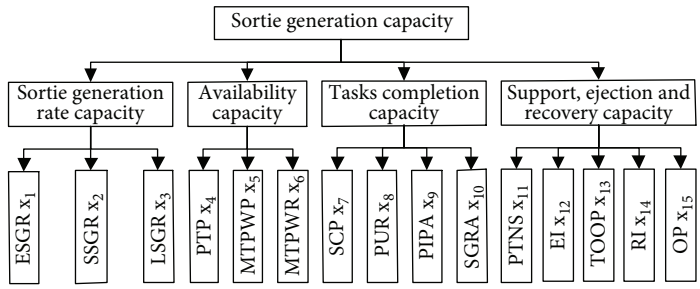


Figure 1: Index system for sortie generation of carrier aircrafts.

ejection interval time, and recovery interval time. So the corresponding high-level indexes will not be included in the evaluation system

(2) Grasping the main indexes makes the evaluation system easy to understand and operate. Environmental factors have little impact on most operations, and they are ignored

(3) The main purpose of this paper is to carry out quantitative evaluation. The indexes which are difficult to quantify, such as the quality of the environment and the ability of personnel, are not used to establish the evaluation system in this paper

In summary, the index system for sortie generation capacity of carrier aircrafts is established with related research results [2]. A three-level index system with complexity, hierarchy, contradiction, and relevance is established by the recursive hierarchy method. The index system for sortie generation capacity of carrier aircrafts is shown in Figure 1.

These indexes are defined as follow:

(1) Emergency sortie generation rate (ESGR): the maximum number of ready aircrafts taking off in a few minutes

(2) Surge sortie generation rate (SSGR): the average number of aircrafts per day in the surge operation (4 days)

(3) Last sortie generation rate (LSGR): the average number of aircrafts per day in the continuous operation (30 days)

(4) Performing tasks proportion (PTP): the time proportion that the aircrafts can carry out one task at least under a certain flight plan and logistics condition

(5) Missing tasks proportion waiting for parts (MTPWP): the proportion of aircrafts missing the tasks due to waiting for parts

(6) Missing tasks proportion waiting for repair (MTPWR): the proportion of aircrafts missing the tasks due to waiting for repair

(7) Scheduled completion proportion (SCP): the proportion of the completed number in the planned number of aircrafts

(8) Pilot utilization rate (PUR): the average utilization rate of the pilots per day

(9) Plan implementation probability per aircraft (PIPA): the plan implementation probability per aircraft under the certain constraints in a given period of time

(10) Sortie generation rate per aircraft (SGRA): the sortie generation rate per aircraft under the certain constraints

(11) Preparation time for next sortie (PTNS): the preparation time for next sortie under the condition of a certain resource allocation

(12) Ejection interval (EI): the average time for ejecting a single aircraft per catapult

(13) Take-off outage proportion (TOOP): the proportion of the cancelled number in the ready number of aircrafts

(14) Recovery interval (RI): the average time for recovering a single aircraft

(15) Overshoot proportion (OP): the proportion of the number of aircrafts failed to recover in the number of aircrafts ready to recover

In the three-level recursive hierarchical graph of Figure 1, there are interactions between the underlying indexes:

(1) In practice, the surge sortie generation rate and last sortie generation rate are contradictory, and they cannot reach the optimal value at the same time

(2) The preparation time for next sortie and the ejection interval constitute one wave duration. If the preparation time is sufficient, the time spent on ejection and recovery will be reduced. And the sortie generation capability will be reduced. If the time spent on ejection and recovery is sufficient, the preparation time may not be able to meet the carrier aircraft support operations, resulting in the reduction of the available carrier aircraft and the reduction of sortie generation capability

Therefore, there are correlations and contradictions among the indexes of sortie generation capability. The

evaluation of sortie generation capability shows complex nonlinear relationship. Using the nonlinear mapping ability of the neural network to evaluate the complex nonlinear sortie generation capability can avoid the subjectivity of traditional evaluation methods and the complexity of the evaluation process.

Some references used different methods to evaluate sortie generation capacity of carrier aircrafts. Xie et al. [21] investigated the complicated relation between the sortie generation and aviation maintenance of the Nimitz class carrier aircraft. It was observed that the state transition diagram was both effective and efficient for system analysis. Liu et al. [22] proposed that sortie generation was one of the critical indexes which were used to characterize carrier and air wing capabilities. In order to research the index system of sortie generation capacity and their effects on the embarked air wings, attention was drawn to the analysis on the basic concept of sortie generation rate (SGR) and a range of constraints for launch and recovery, followed by a review of the definition of the SGR index system that commonly applies in the assessment of various types of carrier aircrafts commissioned in the foreign navies (e.g., the USN, the RN, the French Navy, and the Russian Navy). Analyses were also performed on the SGR index system used for these typical carriers. The conclusion was that the establishment of an effective index system and setting for the relevant indexes must be accomplished by actual operations and combat exercises. Zhou et al. [23] considered that operational capability of the flight deck was the critical factor to affect the sortie generation of a carrier-based aircraft, including launch operation, recovery and repot operations, and serving. The definition of an optimized flight deck operation plan was given. A method to calculate the number of sortie generation in the optimized flight deck operation plan was proposed. Some factors including aircraft number, launch time, repotted time, recovery time, and how they affect the sortie generation were analysed. The results showed that promoting the capability of the flight deck was the key factor to increase the number of sorties. Wang and Yan [24] summarized three evaluation methods of SGR of embarked aircrafts according to the relevant exercise data and papers released by the US Navy. And the characteristics of them were analysed. The advantages of statistical analysis were that the accuracy and credibility of the data were high. The disadvantage was that it can only evaluate the carrier aircraft in service. The cost was high, and the cycle was long. The empirical formula method was based on the recovery data of different carrier aircrafts at different times. It could quickly predict the capacity of the carrier aircraft under typical combat tasks. But there must be a large number of the actual operational data of the carrier aircraft. And the errors of evaluation results were large. The experimental method is based on the operation process of the carrier aircraft. And the computer simulation method was used. It was characterized by wide applicability. The evaluation was of high accuracy, less cost, and short cycle. Zhang et al. [25] established a system with models based on parameters to analyse effectiveness about major factors of the swarming aircraft. The AHP method was used to calculate the weight of each parameter. The fuzzy synthetic evaluation method was applied to access the operational performance. The result demonstrated that the method was feasible to deliver a scientific evaluation and presented a new perspective to judge the efficiency of swarming aircrafts. References [26–28] used the AHP method to evaluate antiship combat capability of the carrier aircraft, the threat the of carrier-borne aircraft, and effectiveness of the carrier-based aircraft in air defence.

3. VS-RBF Neural Network with the Fast Learning Rate

The robust regression method is used to replace the linear least square algorithm to reduce the influence of outliers on weight training. Then, the fast learning rate method is used to adjust the learning rate of RBF neural network, which can guarantee the stable learning of the network and the convergence speed. Finally, the proposed VS-RBF has better nonlinear approximation ability, faster training speed, and a more compact network structure compared with SORBF and RS-RBF.

3.1. Structure of the RBF Neural Network. The structure of the RBF neural network is similar to that of the multilayer forward network. There are three layers: input layer, hidden layer, and output layer. The topology of the RBF neural network is shown in Figure 2.

Therefore, the output y can be described as (1) for multi-input and single-output RBF neural networks,

$$y = \sum_{i=1}^{m} w_i \phi_i(X) = \sum_{i=1}^{m} w_i e^{-\|X-C_i\|^2/2\delta_i^2}, \quad (1)$$

where $X = (x_1, x_2, \cdots, x_n)^T \in R^n$ is the input vector; n is the node number in the input layer; $W = (w_1, w_2, \cdots, w_m)^T \in R^m$ is the weight of the output layer; m is the neuron number in the hidden layer; $\Phi = (\phi_1, \phi_2, \cdots, \phi_m)^T \in R^m$ is radial basis function in the hidden layer, which is the Gauss function, namely, $\phi_i(X) = e^{-\|X-C_i\|^2/2\delta_i^2}$; δ_i is the expansion constant of radial basis function; $\|*\|$ is the Euclid norm; C_i is the data center of ith hidden nodes; and y is the output of the RBF network. $i = 1, 2, \ldots, m$. In Figure 2, the output information (OI) is $\phi_i(X) = e^{-\|X-C_i\|^2/2\delta_i^2}$, and the multi-information (MI) is the intensity of information between C_i and y.

In the human cerebral cortex, the local accommodation and overlapping receptive fields are the characteristics of human brain response. Φ has a strong response in a part of the C_i surrounding area, which reflects the characteristics of the cerebral cortex response. In this paper, the OI intensity of neurons in the hidden layer and the MI intensity between neurons are used to adjust the neurons in the hidden layer. Thus, the topology structure of the neural network is modified.

There are many methods to change the RBF structure. Zheng et al. [29] proposed a meshfree or meshless local RBF collocation method to calculate the band structures of two-dimensional antiplane transverse elastic waves in phononic crystals. Three new techniques were developed for

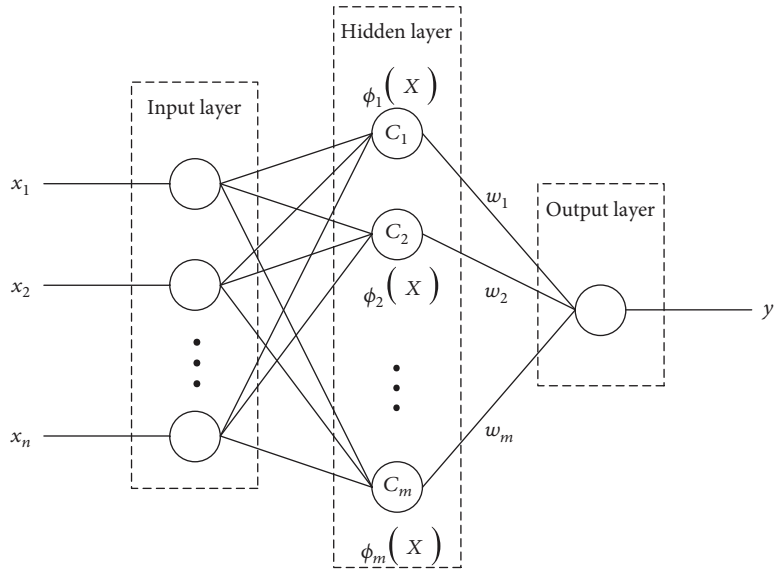


Figure 2: Topology structure of the RBF neural network.

calculating the normal derivative of the field quantity required by the treatment of the boundary conditions, which improved the stability of the local RBF collocation method significantly. The pruning was at the end of the training rather than in the learning process. Therefore, it would be limited in application. Sarimveis et al. [30] presented a new method for extracting the valuable process information from input/output data. The proposed methodology produced dynamical RBF neural network models based on a specially designed genetic algorithm (GA), which was used to autoconfigure the structure of the networks and obtain the model parameters. But this algorithm was a global optimization algorithm, which took longer time in the training process. Han et al. [31] presented a flexible structure RBF neural network (FS-RBFNN). The FS-RBFNN could vary its structure dynamically in order to maintain the prediction accuracy. The hidden neurons in the RBF neural network could be added or removed online based on the neuron activity and mutual information, to achieve the appropriate network complexity and maintain overall computational efficiency. However, the algorithm ignored the implicit relationship between neurons, which might cause an overfitting phenomenon. Wu et al. [32] adopted the cloud RBF neural network as the function approximation structure of approximate dynamic programming, and it had the advantage of the fuzziness and randomness of the cloud model. But the cloud method was a global search algorithm, which would reduce the overall learning speed. Fu and Wang [33] proposed a novel separability-correlation measure (SCM) document to rank the importance of attributes. According to the attribute ranking results, different attribute subsets were used as inputs to a classifier, such as an RBF neural network. Those attributes that increase the validation error were deemed irrelevant and were deleted. The complexity of the classifier could thus be reduced, and its classification performance improved. But the initial values were set with the global sample data, which were hard to obtain in practical applications. Peng et al. [34] proposed a novel hybrid forward algorithm (HFA) for the construction of RBF networks with tunable nodes. The set neural main objective was to efficiently and effectively produce a parsimonious RBF neural network that generalizes well. It was achieved through simultaneous network structure determination and parameter optimization on the continuous parameter space. However, the parameters of this method were too complex. Han et al. [35] proposed a new growing and pruning algorithm for RBF neural network structure design, which was named as self-organizing RBF (SORBF). The growing and pruning algorithm was used to design the structure of the RBF neural network automatically. But the overall training time of SORBF was too long. Ding et al. [36] combined rough set theory with neural network (RS-RBF). The model overcame the shortcoming that when neural network inputs too many dimensions, the structure of the network was too big. In order to solve the problems above, this paper proposes a variable structure RBF neural network (VS-RBF) with a fast learning rate. The number of neurons in the hidden layer is adjusted by calculating the output information (OI) of neurons in the hidden layer and the multi-information (MI) between neurons in the hidden layer and output layer. The convergence of the final network in the structural adjustment process is proved. In this paper, the proposed VS-RBF neural network can be used to grow or prune the neurons in the hidden layer according to the actual system.

3.2. VS-RBF Neural Network. VS-RBF is used to adjust its structure based on information intensity. Firstly, the activity of the neurons in the hidden layer is determined by OI intensity of neurons in the hidden layer. And the neurons with strong activity are decomposed. Secondly, the connection strength between the neurons in the hidden layer and in output layers is analyzed by calculating MI intensity between the neurons. Then the network structure is modified according to MI intensity. Finally, the parameters of the neural network are adjusted. Therefore, the structure of the RBF neural network can be divided into two parts: the decomposition of

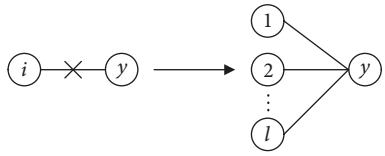


Figure 3: Decomposition of the neuron in the hidden layer.

the neurons in the hidden layer and the interconnection adjustment between the neurons in the hidden layer and in the output layer.

3.2.1. Decomposition of Neurons in the Hidden Layer. The activity of the neuron denotes the information capacity of the neuron. When the neuron provides higher activities, it means that the neuron includes more information. In order to increase the accuracy of RBF, the information should be evenly distributed in neurons. Thus, if the neuron provides higher activities, it should be decomposed to higher number of neurons.

The activity of neurons in the hidden layer is calculated by

$$A_i(X) = \frac{\phi_i}{\sum_{j=1}^{m} \phi_j (\|X - C_i\| + \alpha)}, \tag{2}$$

where $i, j = 1, 2, \ldots, m$, $A_i(X)$ is the activity of the ith neuron in the hidden layer, m is the number of neurons in the hidden layer, ϕ_i is the output of the ith neuron in the hidden layer, ϕ_j is the output of the jth neuron in the hidden layer, and α is a small real number. The activity of neurons in the hidden layer is inversely proportional to the Euclid distance between X and C_i. The closer the distance is, the higher the activity value is. The activity of neurons $A_i(X)$ is obtained from the output of the hidden layer, which denotes the need of decomposition.

When activity $A_i(X)$ of the neuron in the hidden layer is strong, the neuron is decomposed, and the neuron is decomposed to l new neurons. The main idea behind this decomposition is to evenly distribute information in neurons in order to increase the accuracy of VS-RBF. When the activity of the ith neuron in the hidden layer is greater than the threshold of activity $A_0 = \max(100e_d, 1/m)$ (e_d is the expected error), the connection between the ith neuron and the output neuron is broken down as shown in Figure 3.

In Figure 3, if the activity between the ith neuron and the output neuron y is greater than the threshold of activity A_0, the connection between the ith neuron and the output neuron y is broken down. After decomposition, the ith neuron is decomposed to l new neurons, and each of l new neurons connects to the output neuron y. l = integer $(A_i/(\sum_{i=1}^{m} A_i(X)/m))$.

After the connection between the ith neuron and the output neuron is broken down, there will be l new neurons connecting to the output neuron. The initial center and variance of the new neurons are C_{ij} and δ_{ij}, which can be obtained from

$$C_{ij} = \lambda_i C_i + \mu_i X, \tag{3}$$

$$\delta_{ij} = \lambda_i \delta_i, \tag{4}$$

where $0.9 \le \lambda_i \le 1.1$, $0 \le \mu_i \le 0.2$, and C_i and δ_i are the center and variance of the ith neuron in the hidden layer, respectively, and C_{ij} and δ_{ij} are the center and variance of the jth neuron decomposed from the ith neuron. $j = 1, 2, \ldots, l$. l is the number of the new neurons. l is the integer part of A_i/AV. $AV = \sum_{i=1}^{m} A_i(X)/m$.

The weight w_{ij} between the new neuron and the output neuron is

$$w_{ij} = \beta_j \frac{w_i \phi_i(X) - e}{\phi_{ij}(X)}, \tag{5}$$

where β_j is the decomposition parameter of the jth neuron; $\sum_{j=1}^{l} \beta_j = 1$, $\beta_j = 1/l$; $\phi_{ij}(X)$ is the output of the new neuron; and e is the output error of the neural network before decomposition.

3.2.2. Interconnection Adjustment between the Neurons in the Hidden Layer and in the Output Layer. The MI function $I(a; y)$ of neuron a in the hidden layer and neuron y in the output layer is obtained from (6). $I(a; y)$ depends on the intensity of information between a and y,

$$I(a; y) = \sum_{a,y} f(a; y) \log_2 \left[f \frac{(a; y)}{f} (a) f(y) \right], \tag{6}$$

where $f(a; y)$ is the joint distribution density function of a and y. $f(a; y) = f(a) f(y \mid a)$, where $f(a)$ and $f(y)$ are the probability densities of a and y, respectively. According to the Shannon entropy theory, $f(a; y)$, $f(a)$, and $f(y)$ are not calculated in fact, and the entropies are calculated instead.

Assuming that a and y are interconnected neurons, $I(a; y)$ depends on the intensity of information between a and y.

According to the Shannon entropy theory, the connection strength $I(a; y)$ between a and y can be calculated by

$$I(a; y) = \mathrm{EN}(a) - \mathrm{EN}(a|y) = \mathrm{EN}(y) - \mathrm{EN}(y|a), \tag{7}$$

where $\mathrm{EN}(a)$ is the entropy of a and $\mathrm{EN}(a \mid y)$ is the entropy of a in the condition of y. When a and y are independent, $I(a; y) = 0$. Otherwise, $I(a; y) > 0$. Therefore, $I(a; y) \ge 0$. The range of $I(a; y)$ is shown in

$$I(a; y) \le \min(\mathrm{EN}(a), \mathrm{EN}(y)). \tag{8}$$

Then, based on (8), the normalized MI is obtained from

$$I_s(a; y) = I(a; y) / \min(\mathrm{EN}(a), \mathrm{EN}(y)), \tag{9}$$

where $I_s(a; y) \in [0, 1]$. The calculation of I_s is able to determine the interaction strength between a and y.

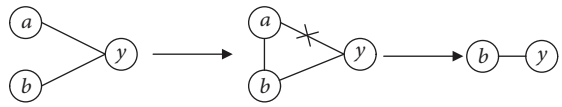


Figure 4: Disconnection of the neuron in the hidden layer and the neuron in output layers.

In the RBF neural network, when I_s is large, the interaction strength between a and y is strong and the connection of a and y exists. When I_s approaches 0, it indicates that the interaction strength between a and y is weak. Then the connection of a and y can be deleted in the structure adjustment. It can reduce the redundancy of the neural network.

In Figure 4, when $I_s \le I_0$ ($I_0 = 0.01e_d$), the connection between a and y is disconnected. In the hidden layer, the nearest neuron b with a is found. The new neuron b contains neurons a and b. Thus, a new layer is not added to the network, and there is no weight between neurons a and b. The parameters of neuron b are adjusted from (10)–(12). C_b' is the same as C_a. δ_b' is the same as δ_a:

$$C_b' = C_a, \tag{10}$$

$$\delta_b' = \delta_a, \tag{11}$$

$$w_b' = w_b + w_a \cdot \phi_a \frac{X}{\phi_b}(X), \tag{12}$$

where C_b', δ_b', and w_b' are the center, variance, and weight between b and y in the new output layer, respectively.

VS-RBF can not only increase the number of neurons in the hidden layer but also remove redundant neurons. The optimal number of RBF neurons is obtained by adjusting the output information of neurons in the hidden layer and the multi-information between neurons in the hidden layer and output layer. When the error of output meets the demand, the optimal number of RBF neurons is obtained.

3.3. Robust Regression Training Method for Weights in the Output Layer. At present, RBF neural network weights are usually trained by the linear least square algorithm, but the least square estimation is affected by outliers. In order to reduce the influence of outliers, the robust regression method is applied to train weights in the output layer.

The output of the RBF network is the weighted sum of the output of the hidden layer. The aim of the training is to make the sum of squared error $E(W)$ between the output of the whole network and the actual output least. The sum of squared error $E(W)$ is obtained from

$$E(W) = \min \sum_{t=1}^{N} \left(y_t - \Phi_t^T W\right)^2 = \min \sum_{t=1}^{N} \sigma_t^2, \tag{13}$$

where y_t is the value of actual output; Φ_t^T is the output sample vector of the hidden layer; $W(t)$ is the weight vector; N is the number of samples; and $t = 1, 2, \ldots, N$.

In order to reduce the influence of outliers, we hope to find a function $\theta(\sigma_t)$, which increases with the increase of σ_t and the growth rate is slower than σ_t^2. Then the sum of squared error $E(W)$ can be expressed as

$$E(W) = \min \sum_{t=1}^{N} \theta(\sigma_t). \tag{14}$$

$\theta(\sigma_t)$ is an Andrews function, which can be expressed as (15). And the derivative of $\theta(\sigma_t)$ can be expressed as

$$\theta(\sigma_t) = \begin{cases} K^2 - K^2 \cos \dfrac{\sigma_t}{K}, & \sigma_t \ge \pi K, \\ 2K^2, & \sigma_t < \pi K, \end{cases} \tag{15}$$

$$\theta'(\sigma_t) = \begin{cases} K \sin \dfrac{\sigma_t}{K}, & \sigma_t \ge \pi K, \\ 0, & \sigma_t < \pi K, \end{cases} \tag{16}$$

where $\theta'(\sigma_t)$ is the derivative of $\theta(\sigma_t)$ and K is a threshold constant.

Therefore, robust regression can be written in recursive form, which is expressed in

$$W(t) = W(t-1) - \eta \frac{\partial \theta(\sigma_t)}{\partial W}, \tag{17}$$

where W is the estimated weight and η is the training coefficient, which can be determined according to the actual situation. Then $W(t)$ is transferred to

$$\begin{aligned} W(t) &= W(t-1) - \eta \frac{\partial \theta(\sigma_t)}{\partial \sigma_t} \cdot \frac{\partial \sigma_t}{\partial W} \\ &= W(t-1) - \eta \theta'(\sigma_t) \cdot \left(-\Phi_t^T\right) \\ &= W(t-1) + \eta \theta' \left(y_t - \Phi_t^T W(t)\right) \Phi_t^T. \end{aligned} \tag{18}$$

It can reduce the influence of outliers on neural network training by using the robust regression method.

3.4. Fast Learning Rate. In the use of the RBF neural network, the learning rate was often subjective to set as a fixed value. It remained unchanged throughout the learning process. If the learning rate was set too high, the convergence speed of the network might be very fast and it might cause network instability. If the learning rate was too small, it would cause that the network convergence speed was slow and consume a large amount of computing time. Therefore, it is very difficult to choose a suitable learning rate for the traditional RBF neural network. In order to solve this problem, this paper proposes a new fast learning rate. The fast learning rate is suitable for each step of the iteration, which can ensure the stability of the network. At the same time, the convergence speed of the network and the efficiency of the network can be improved.

Set

$$\Phi = \begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1m} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{N1} & \phi_{N2} & \cdots & \phi_{Nm} \end{bmatrix}. \tag{19}$$

N is the number of samples. m is the number of nodes in the hidden layer. $\widehat{y}$ is the output calculated by the network. Then the cost function $E(t)$ of the tth train can be defined in

$$E(t) = \frac{1}{2}\sum_{j=1}^{N}\left(y_j(t) - \widehat{y}_j(t)\right)^2 = \frac{1}{2}\sum_{j=1}^{N} e_j^2(t). \tag{20}$$

The output error is $e(t) = [e_1(t), e_2(t), \cdots, e_N(t)]^T$. $\Delta W(t) = W(t) - W(t-1)$ is the change of the weight in the tth train. According to (18), $\Delta W(t)$ is obtained in

$$\Delta W(t) = \eta(t)\theta'\left(y_t - \Phi_t^T W(t)\right)\Phi_t^T. \tag{21}$$

The increase of error can be expressed as

$$\begin{aligned} \Delta e(t) &= e(t) - e(t-1) \\ &= y(t) - \widehat{y}(t) - y(t-1) + \widehat{y}(t-1). \end{aligned} \tag{22}$$

$\Delta y(t) = y(t) - y(t-1)$ is the change of the actual output. $\Delta\widehat{y}(t) = \widehat{y}(t) - \widehat{y}(t-1)$ is the change of the network output. In general, the absolute value of the change of the actual output is far less than the absolute value of the change of the network output. $|\Delta y(t)| \ll |\Delta\widehat{y}(t)|$. The change of the actual output is negligible compared with the change of the network output, because the actual output is often constrained by many conditions, and the network output will not be limited. This assumption has practical significance. Thus, (22) can be approximated as

$$\Delta e(t) = e(t) - e(t-1) \approx -\Delta\widehat{y}(t). \tag{23}$$

Considering (21), the change of error $\Delta e(t)$ is expressed in

$$\begin{aligned} \Delta e(t) &\approx -\Delta\widehat{y}(t) = -\Phi\Delta W(t) \\ &= -\eta(t)\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T. \end{aligned} \tag{24}$$

Then, $e(t)$ is expressed in (25) considering (24):

$$\begin{aligned} e(t) &= e(t-1) + \Delta e(t) \\ &\approx e(t-1) - \eta(t)\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T. \end{aligned} \tag{25}$$

The cost function $E(t)$ of the tth train can be obtained from (25). $E(t)$ is expressed in

$$\begin{aligned} E(t) &= \frac{1}{2}e^T(t)e(t) \\ &\approx \frac{1}{2}\left[e(t-1) - \eta(t)\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T\right]^T \\ &\quad \cdot \left[e(t-1) - \eta(t)\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T\right]. \end{aligned} \tag{26}$$

This cost function can be considered as a function of the learning rate $\eta(t)$. The optimal value of learning rate $\eta(t)$ can be obtained by minimizing $E(t)$. The first order condition of (26) is expressed in

$$\begin{aligned} \frac{\partial E(t)}{\partial \eta(t)}\Big|_{\eta(t)=\eta^*(t)} &= -\frac{1}{2}\left[\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T\right]^T \\ &\quad \cdot \left[e(t-1) - \eta(t)\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T\right] \\ &\quad - \frac{1}{2}\left[e(t-1) - \eta(t)\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T\right]^T \\ &\quad \cdot \Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T = 0. \end{aligned} \tag{27}$$

The second order condition of (26) is expressed in

$$\begin{aligned} &\frac{\partial^2 E(t)}{\partial \eta^2(t)}\Big|_{\eta(t)=\eta^*(t)} \\ &= \frac{1}{2}\left[\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T\right]^T \Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T \\ &\quad + \frac{1}{2}\left[\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T\right]^T \Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T \\ &= \left[\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T\right]^T \Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T > 0. \end{aligned} \tag{28}$$

The second order condition (28) holds as Φ is positive definite. And the fast learning rate $\eta^*(t)$ can be obtained by (27). $\eta^*(t)$ is expressed in

$$\begin{aligned} \eta^*(t) &= \frac{1}{2}\frac{\left[\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T\right]^T e(t-1)}{\left[\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T\right]^T \Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T} \\ &\quad + \frac{1}{2}\frac{e^T(t-1)\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T}{\left[\Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T\right]^T \Phi\theta'\left(y - \Phi^T W(t)\right)\Phi^T}. \end{aligned} \tag{29}$$

3.5. Learning Algorithm of VS-RBF with the Fast Learning Rate. VS-RBF with the fast learning rate can not only increase the number of neurons in the hidden layer but also remove redundant neurons. At the same time, the use of the robust regression algorithm can avoid the influence of the abnormal sample on the neural network. And the fast learning rate can ensure that the learning rate is appropriate in the iterative process. In this paper, the steps of the learning algorithm are shown in Figure 5.

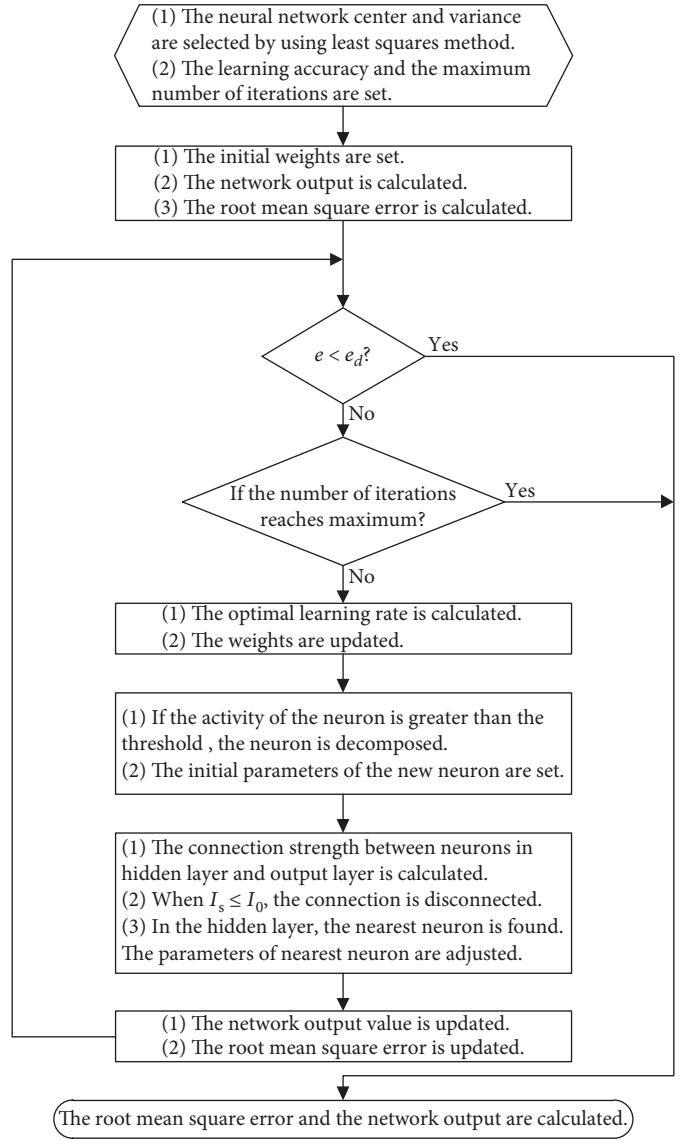


Figure 5: Learning algorithm of VS-RBF with the fast learning rate.

Step 1. For a given RBF neural network, the number of neurons in the hidden layer is a small natural number. Considering that VS-RBF can adjust the number of neurons in the hidden layer, the number of initial neurons in the hidden layer is defined as the neurons in the input layer or a little more. In this paper, we consider that the number of initial neurons in the hidden layer is equal to the neurons in the input layer. The neural network center C and variance δ are selected by using the least square method. Φ is calculated. The learning accuracy e_d and the maximum number of iterations max T are set. The number of neurons in the input layer is the number of input. The number of neurons in the output layer is the number of output. $A_0 = \max(100e_d, 1/m)$, $I_0 = 0.01e_d$

Step 2. The initial weights between the hidden layer and the output layer W_0 are set. The network output $\hat{y} = \Phi W$ is calculated. And the root mean square error e between the actual output and the network output is calculated

Step 3. If $e < e_d$ or the number of iterations reaches max T, go to Step 7. Otherwise, the fast learning rate $\eta^*(t)$ is calculated by (29). The weights are updated by $W(t) = W(t-1) + \eta^*(t)\theta'(y_t - \Phi_t^T W(t))\Phi_t^T$

Step 4. The activity $A_i(X)(i = 1, 2, \cdots, m)$ of the neuron in the hidden layer is calculated. If $A_i(X)$ is greater than the threshold A_0, the neuron i is decomposed, and the network structure is adjusted. The initial parameters of the new neuron are set according to (3) and (4)

Step 5. The connection strength I_s between neuron a in the hidden layer and neuron y in the output layer is calculated. When $I_s \le I_0$, the connection of a and y is disconnected. In the hidden layer, the nearest neuron b with a is found. The parameters of neuron b are adjusted as (10)–(12)

Step 6. The network output value $\hat{y} = \Phi W$ is updated. The root mean square error between the actual output and the network output e is updated. Return to Step 3

Step 7. The root mean square error e and the network output $\hat{y}$ are calculated

VS-RBF with fast learning rate can realize the self-organization of the structure and determine whether to grow or prune the neurons in the hidden layer by calculating the information intensity. A new RBF neural network structure design method is proposed. It not only can adjust the weights of neural network online but also can grow or prune the neurons in the hidden layer. From the biological point of view, this neural network structure is more similar to the mechanism of human brain neuron information processing.

3.6. Comparison of VS-RBF, SORBF, and RS-RBF. The advantages, main novelties, and disadvantages of SORBF [35] and RS-RBF [36] are as follows:

(1) SORBF: neither the number of nodes in the hidden layer nor the parameters need to be predefined and fixed. They are adjusted automatically in the learning process by comparing $\|v_x - P(k)\| = \min_{x=1,2,\ldots,N} \|v_x - P(k)\|$ and βr_x. This type of SORBF-based approach offers a promisingly inexpensive approach to real-time measurement of variables that have typically proved difficult to measure reliably using hardware. However, this algorithm ignores the adjustment of structural parameters, which leads to the slow convergence speed of the neural network learning algorithm

(2) RS-RBF: by processing multiple nodes of the network at one time, multiple hidden nodes can be cut off, and the core nodes in the hidden nodes can be found by calculating the output of the network as $D_n = (\text{output}_n - \text{OUTPUT})/\text{OUTPUT}$. Adaptive principle is introduced to make the segmentation change with pruning. The memory function is to remember the most important nodes for each pruning, and the most important nodes will be not deleted in the subsequent pruning even if the output of these nodes is small. However, this method was affected by the initial value, and sometimes, the final RBF neural network was unstable

4. Proof of Convergence

The convergence of VS-RBF with the fast learning rate affects the performance of the final network.

4.1. Fixed Neural Network Structure. When the neural network structure is fixed, it can be concluded that VS-RBF with the fast learning rate can guarantee the convergence of the final neural network referring to [31].

Theorem 1. *According to the changed neural network structure of VS-RBF, the error after decomposition is 0, and the error after interconnection adjustment is equal to the error of the fixed neural network structure.*

4.2. Changed Neural Network Structure. At the moment t, VS-RBF with the fast learning rate has m neurons in the hidden layer. Current error is $e_m(t)$.

4.2.1. Decomposition of the Neuron in the Hidden Layer. When the neuron a in the hidden layer is decomposed, the number of new neurons is l after decomposition. Then the number of neurons in the hidden layer becomes $m + l - 1$ after decomposition. The error after decomposition $e'_{m+l-1}(t)$ is expressed in

$$\begin{aligned} e'_{m+l-1}(t) &= \sum_{i=1}^{m+l-1} w_i \phi_i(X(t)) - y_d(t) \\ &= \sum_{i=1}^{m} w_i \phi_i(X(t)) + \sum_{j=1}^{l} w_{a,j} \phi_j(X(t)) \\ &\quad - w_a \phi_a(X(t)) - y_d(t), \end{aligned} \tag{30}$$

where $y_d(t)$ is expected output at the moment t and $X(t)$ is a sample at the moment t. It can be obtained from (3) and (4).

In (30), two terms in the middle can be expressed in Eq. (31). $\sum_{j=1}^{l} \beta_j = 1$,

$$\begin{aligned} &\sum_{j=1}^{l} w_{a,j} \phi_j(X(t)) - w_a \phi_a(X(t)) \\ &= \sum_{j=1}^{l} \beta_j \frac{w_a \phi_a(X(t)) - e_m(t)}{\phi_j(X(t))} \phi_j(X(t)) - w_a \phi_a(X(t)) \\ &= \sum_{j=1}^{l} \beta_j (w_a \phi_a(X(t)) - e_m(t)) - w_a \phi_a(X(t)) \\ &= w_a \phi_a(X(t)) - e_m(t) - w_a \phi_a(X(t)) = -e_m(t), \end{aligned} \tag{31}$$

where $X(t) = (x_1, x_2, \cdots, x_n)^T \in R^n$ is the input vector.

According to (30) and (31), the error of the neural network after decomposition $e'_{m+l-1}(t)$ is expressed in

$$\begin{aligned} e'_{m+l-1}(t) &= \sum_{i=1}^{m} w_i \phi_i(X(t)) - y_d(t) - e_m(t) \\ &= e_m(t) - e_m(t) = 0. \end{aligned} \tag{32}$$

After adjusting the structure, the error of network output at the time t is zero. The convergence of average error E is speeded up ($E = 1/2N\sum_{i=1}^{N} e^2(i)$, N is the number of samples). It reflects the decomposition of neurons can improve neural network learning efficiency.

4.2.2. Interconnection Adjustment between the Neurons in the Hidden Layer and in the Output Layer. At the moment t, the connections between the neuron a in the hidden layer and neuron y in the output layer need to be disconnected. And the nearest neuron with a is b. Then error of network output $e'_{m-1}(t)$ will become (33) after disconnection,

$$e'_{m-1}(t) = \sum_{i=1}^{m} w_i \phi_i(X(t)) - y_d(t) - w_a \phi_a(X(t)). \tag{33}$$

According to (10)–(12), (33) can be rewritten as

$$\begin{aligned} e'_{m-1}(t) &= \sum_{i=1, i\neq b}^{m} w_i \phi_i(X(t)) - y_d(t) \\ &\quad - w_a \phi_a(X(t)) + w'_b \phi_b(X(t)) \\ &= \sum_{i=1, i\neq b}^{m} w_i \phi_i(X(t)) - y_d(t) - w_a \phi_a(X(t)) \\ &\quad + \left(w_b + w_a \frac{\phi_a(X(t))}{\phi_b(X(t))} \right) \phi_b(X(t)) \\ &= \sum_{i=1, i\neq b}^{m} w_i \phi_i(X(t)) - y_d(t) - w_a \phi_a(X(t)) \\ &\quad + w_b \phi_b(X(t)) + w_a \phi_a(X(t)) \\ &= \sum_{i=1}^{m} w_i \phi_i(X(t)) - y_d(t) = e_m(t). \end{aligned} \tag{34}$$

Therefore, the disconnection of the neurons in the hidden layer and the neurons in output layers cannot affect the error of network output. Thus, the optimization process of the RBF structure does not affect the convergence of the neural network.

To sum up, VS-RBF with the fast learning rate can guarantee the convergence of the final network. At the same time, the algorithm is simple. The neural network can not only realize the structure and parameter adjustment but also consider the weight changes in the structure of the optimization process.

5. Experimental Validation

VS-RBF with the fast learning rate can adjust the number of neurons in the hidden layer on the basis of the complexity of the object and improve the performance of the RBF neural network. Compared with other self-organizing RBF neural networks (SORBF and RS-RBF), VS-RBF has a more compact structure, faster dynamic response speed, and better generalization ability in approximating a typical nonlinear function, identifying UCI datasets, and evaluating sortie generation capacity of the carrier aircraft.

5.1. Nonlinear Function Approximation. The nonlinear function SIF is selected as

$$y = 2x_1^2 \sin(4x_1) + x_2^2 + \sin(3x_2) + x_1 \sin(4x_2), \tag{35}$$

where $x_1 \in [-2, 2]$ and $x_2 \in [-2, 2]$. The nonlinear function SIF is commonly used to test the performance of the neural network [37].

Data points are randomly sampled adding white Gaussian noise with a standard deviation of 0.01 to produce training and validation datasets, each containing 300 samples. Thus, 600 groups of samples are selected. 300 groups are used for training and the other 300 groups are used to test. The simulation environment is a computer with Intel Core i3–4160 CPU, 4.00 GB RAM and 64 bit operation system. The software is Matlab R2016b.

The main steps on the identification of VS-RBF are as follows:

Step 1. The structure of the neural network is 2-3-1. The initial number of neurons in the hidden layer is set as 3. The neural network center C is selected as [−2, 0, −2; −2, 0, −2]. The variance δ is 3. The initial function width is 1. The learning accuracy e_d is 0.01. The maximum number of iterations max T is set as 10,000

Step 2. The initial weights between the hidden layer and the output layer W_0 are set as 1. The network output $\hat{y} = \Phi W$ is calculated. And the root mean square error e between the actual output and the network output is calculated

Step 3. If $e < e_d$ or the number of iterations reaches max T, go to Step 7. Otherwise, the fast learning rate $\eta^*(t)$ is calculated by (29). The weights are updated by $W(t) = W(t-1) + \eta^*(t)\theta'(y_t - \Phi_t^T W(t))\Phi_t^T$

Step 4. The activity $A_i(X)(i = 1, 2, \cdots, m)$ of the neuron in the hidden layer are calculated. If $A_i(X)$ is greater than the threshold A_0, the neuron i is decomposed, and the network structure is adjusted. The initial parameters of the new neuron are set according to (3) and (4)

Step 5. The connection strength I_s between neuron a in the hidden layer and neuron y in the output layer is calculated. When $I_s \leq I_0$, the connection of a and y is disconnected. In the hidden layer, the nearest neuron b with a is found. The parameters of neuron b are adjusted as (10)–(12)

Step 6. The network output value $\hat{y} = \Phi W$ is updated. The root mean square error between the actual output and the network output e is updated. Return to Step 3

Table 1: Comparisons of performances in approximating SIF.

Algorithm	Learning accuracy	Actual error	Node number of the final hidden layer	Training time (s)
VS-RBF	0.01	0.0148	14	213.41
SORBF	0.01	0.0164	16	221.37
RS-RBF	0.01	0.0206	21	459.12

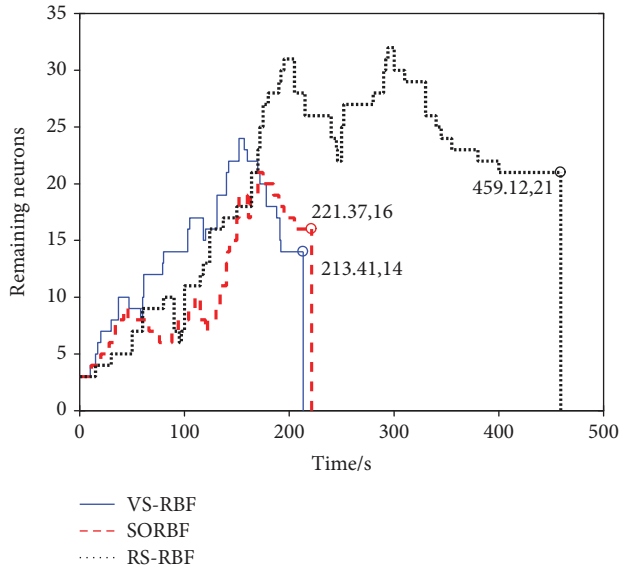


Figure 6: Number of remaining neurons in SIF training.

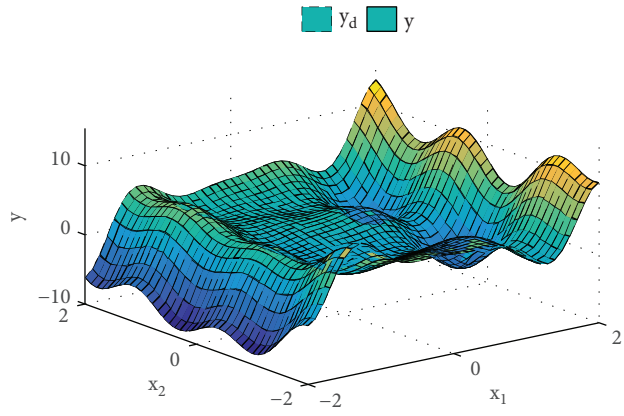


Figure 7: Effect of VS-RBF in SIF approximation.

Step 7. The root mean square error e and the network output $\hat{y}$ are calculated

Under this condition, the comparison of performances of VS-RBF, SORBF [35], and RS-RBF [36] is shown in Table 1. The remaining neurons in the hidden layer in the SIF approximation are shown in Figure 6. The approximation effect of SIF is shown in Figure 7. The error surface is shown in Figure 8.

Figure 6 shows the changes of the number of remaining neurons in the training process. We can find that the structure adjustment of VS-RBF is stable, and the structure is most compact. A neuron of VS-RBF can be decomposed into several at one time. VS-RBF structure adjustment is quicker. The information processing ability of the RBF neural network is improved.

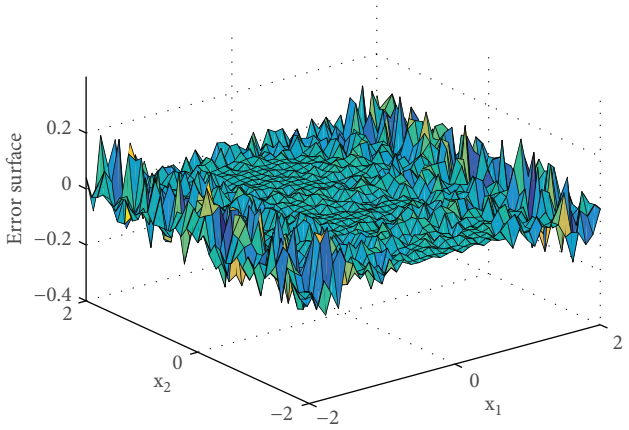


Figure 8: SIF approximation error of VS-RBF.

Figure 7 shows that VS-RBF can well approximate the nonlinear function SIF after training. And the output value of VS-RBF coincides with the actual value.

Figure 8 shows the error surface of VS-RBF in approximation. The test error is less than 0.015. Table 1 gives the comparison of VS-RBF, SORBF, and RS-RBF. Under the same initial conditions, the training times of SORBF and RS-RBF are more than that of VS-RBF. The structures of SORBF and RS-RBF after training are more complex than that of VS-RBF.

In addition, when using the trained neural network for function approximation, the test errors of SORBF and RS-RBF are larger than that of VS-RBF. The VS-RBF neural network has faster training speed, more compact network structure, and stronger nonlinear function approximation ability.

5.2. UCI Datasets. In order to show the effectiveness of VS-RBF, the identification effect is justified on UCI datasets. Istanbul Stock Exchange Dataset is selected in UCI datasets. Data is collected from http://imkb.gov.tr and http://finance .yahoo.com. Data is organized with regard to working days in Istanbul Stock Exchange. The selected datasets include returns of Istanbul Stock Exchange (ISE) with seven other international indexes: Standard & Poor's 500 return index (SP), stock market return index of Germany (DAX), stock market return index of UK (FTSE), stock market return index of Japan (NIK), stock market return index of Brazil (BVSP), MSCI European Index (EU), and MSCI Emerging Markets Index (EM) from June 5, 2009, to February 22, 2011. There are 536 groups in this dataset. The first 436 groups are used to train the network, and the last 100 groups are used to test the network. The input number is 7, and the output (ISE) number is 1.

The main steps on the identification of VS-RBF on UCI datasets are as follows:

Step 1. The structure of the neural network is 7-10-1. The initial number of neurons in the hidden layer is set as 10. The neural network center C is selected as [−3, −3, −3, −3, −3, −3, −3, −3, −3, −3; −2, −2, −2, −2, −2, −2, −2, −2, −2, −2; −1, −1, −1, −1, −1, −1, −1, −1, −1, −1; 0, 0, 0, 0, 0, 0, 0, 0, 0, 0; 1, 1, 1, 1, 1, 1, 1, 1, 1, 1; 2, 2, 2, 2, 2, 2, 2, 2, 2, 2; 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,]. The variance δ is 2. The initial function width is 2. The learning accuracy e_d is 0.01. The maximum number of iterations max T is set as 10,000

Step 2. The initial weights between the hidden layer and the output layer W_0 are set as 1. The network output $\hat{y} = \Phi W$ is calculated. And the root mean square error e between the actual output and the network output is calculated

Step 3. If $e < e_d$ or the number of iterations reaches max T, go to Step 7. Otherwise, the fast learning rate $\eta^*(t)$ is calculated by (29). The weights are updated by $W(t) = W(t-1) + \eta^*(t)\theta'(y_t - \Phi_t^T W(t))\Phi_t^T$

Step 4. The activity $A_i(X)(i = 1, 2, \cdots, m)$ of the neuron in the hidden layer is calculated. If $A_i(X)$ is greater than the threshold A_0, the neuron i is decomposed, and the network structure is adjusted. The initial parameters of the new neuron are set according to (3) and (4)

Step 5. The connection strength I_s between neuron a in the hidden layer and neuron y in the output layer is calculated. When $I_s \le I_0$, the connection of a and y is disconnected. In the hidden layer, the nearest neuron b with a is found. The parameters of neuron b are adjusted as (10)–(12)

Step 6. The network output value $\hat{y} = \Phi W$ is updated. The root mean square error between the actual output and the network output e is updated. Return to Step 3

Step 7. The root mean square error e and the network output $\hat{y}$ are calculated

Under this condition, the comparison of performances of VS-RBF, SORBF [35], and RS-RBF [36] is shown in Table 2. The identification effect in the test is shown in Figure 9. The identification error is shown in Figure 10.

In Table 2, the remaining node number in the hidden layer of VS-RBF is the least. And the actual error and training time of VS-RBF are less than those of SORBF and RSRBF.

Figure 9 shows that VS-RBF can well identify the test data. And the output value of VS-RBF coincides with the actual value.

Figure 10 shows the identification error of VS-RBF. The test error is less than 0.014.

Thus, when using the trained neural network for identification, the test errors of SORBF and RS-RBF are larger than that of VS-RBF. The VS-RBF neural network has faster training speed, more compact network structure, and stronger identification ability in Istanbul Stock Exchange Dataset.

Table 2: Comparisons of performances in UCI datasets.

Algorithm	Learning accuracy	Actual error	Node number of the final hidden layer	Training time (s)
VS-RBF	0.01	0.0131	20	230.56
SORBF	0.01	0.0208	24	292.30
RS-RBF	0.01	0.0263	33	441.52

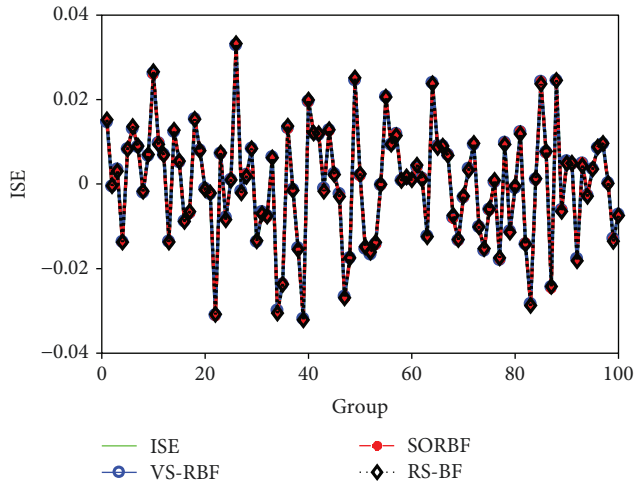


Figure 9: Justification of UCI datasets.

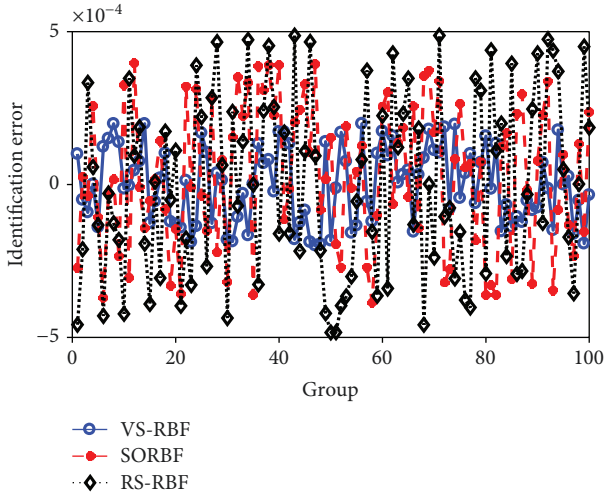


Figure 10: Identification error of UCI datasets.

6. Evaluation for Sortie Generation Capacity of the Carrier Aircraft

The inputs of the neural network are the evaluation indexes of sortie generation capacity of the carrier aircraft. The output of the neural network is the evaluation value of sortie generation capacity of the carrier aircraft by the expert scoring method. This paper selects the surge operation of the "Nimitz" carrier in 1997 as the object [1].

Table 3: Sortie generation rate capacity.

Group	x_1 (sortie)	x_2 (sortie/day)	x_3 (sortie/day)	Evaluation
1	30	250	200	0.8622
2	31	240	180	0.2943
3	29	235	210	0.8112
4	33	260	220	0.5285
5	32	210	170	0.1656
6	29	245	194	0.6020
7	27	267	230	0.9630
8	32	211	183	0.6541
9	25	261	201	0.6892
10	32	232	196	0.5482

Table 4: Aircraft availability capacity.

Group	x_4 (%)	x_5 (%)	x_6 (%)	Evaluation
1	80	11	9	0.8622
2	85	20	5	0.2943
3	90	4	6	0.8112
4	75	11	14	0.5285
5	82	10	18	0.1656
6	91	3	5	0.6020
7	78	23	9	0.9630
8	84	9	7	0.6541
9	85	1	4	0.6892
10	72	2	16	0.5482

Table 5: Task completion capacity.

Group	x_7 (%)	x_8 (sortie/day)	x_9 (%)	x_{10} (sortie/day)	Evaluation
1	85	2.5	90	6	0.8622
2	74	2.2	80	7	0.2943
3	81	2.0	84	5	0.8112
4	90	1.8	75	6	0.5285
5	61	1.5	68	5	0.1656
6	86	1.9	86	8	0.6020
7	78	2.1	88	5	0.9630
8	65	2.3	94	6	0.6541
9	79	2.4	81	7	0.6892
10	83	1.7	82	5	0.5482

Table 6: Support, ejection, and recovery capabilities.

Group	x_{11} (minute)	x_{12} (minute)	x_{13} (%)	x_{14} (minute)	x_{15} (%)	Evaluation
1	30	1	1	1.5	3.3	0.8622
2	32	2	0.5	1.8	5	0.2943
3	28	1.4	1.2	1.4	1	0.8112
4	25	1.6	1.6	1.9	7	0.5285
5	33	2.5	2	2.2	6	0.1656
6	45	1.1	3	1.1	2	0.6020
7	27	0.6	0.6	1.2	10	0.9630
8	26	0.7	0.8	1.6	3	0.6541
9	36	0.5	1.6	1.7	4.5	0.6892
10	29	1.2	0.4	2.1	6	0.5482

500 groups of data are selected excluding abnormal data. 400 groups are used in training, and 100 groups are used in testing. The inputs of the neural network are $x_1 \sim x_{15}$, which are corresponding to the 15 indexes of the evaluation index system, respectively.

The number of output is 1, which value is $[0, 1]$. The data of each group in the 500 groups of samples included 15 normalized lowest level indexes and one evaluation value in the range of $[0, 1]$ for sortie generation capacity of the carrier aircraft evaluated by experts. Due to limitation of space, 10 groups of samples are given in Tables 3–6.When the evaluation for sortie generation capacity of the carrier aircraft is closer to 1, it is indicated that the allocation of the sortie generation scheme under the current indexes is more reasonable, and the sortie generation capacity of the carrier aircraft is higher. Otherwise, when the evaluation for sortie generation capacity of the carrier aircraft is closer to 0, it indicates that the allocation of the sortie generation scheme under the current indexes is more unreasonable. And the sortie generation capacity of the carrier aircraft is lower, which means that the scheme should be advanced.

The main steps on the identification of VS-RBF are as follows:

Step 1. The number of neurons in the hidden layer is set to 15 initially. The structure of neural network is 15-15-1. The initial number of neurons in the hidden layer is set as 15. The neural network center C is selected as $0.1^*[-7, -7, -7, -7, -7, -7, -7, -7, -7, -7, -7, -7, -7, -7, -7; -6, -6, -6, -6, -6, -6, -6, -6, -6, -6, -6, -6, -6, -6, -6; -5, -5, -5, -5, -5, -5, -5, -5, -5, -5, -5, -5, -5, -5, -5; -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -4; -3, -3, -3, -3, -3, -3, -3, -3, -3, -3, -3, -3, -3, -3, -3; -2, -2, -2, -2, -2, -2, -2, -2, -2, -2, -2, -2, -2, -2, -2; -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1; 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0; 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1; 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2; 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3; 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4; 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5; 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6; 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7]$. The variance δ is 0.5. The initial function width is 1. The learning accuracy e_d is 0.001. The maximum number of iterations max T is set as 10,000

Step 2. The initial weights between the hidden layer and the output layer W_0 are set as 0.5. The network output $\hat{y} = \Phi W$ is calculated. And the root mean square error e between the actual output and the network output is calculated

Step 3. If $e < e_d$ or the number of iterations reaches max T, go to Step 7. Otherwise, the fast learning rate $\eta^*(t)$ is calculated by (29). The weights are updated by $W(t) = W(t-1) + \eta^*(t)\theta'(y_t - \Phi_t^T W(t))\Phi_t^T$

Step 4. The activity $A_i(X)(i = 1, 2, \cdots, m)$ of the neuron in the hidden layer is calculated. If $A_i(X)$ is greater than the threshold A_0, the neuron i is decomposed, and the network structure is adjusted. The initial parameters of the new neuron are set according to (3) and (4)

Step 5. The connection strength I_s between neuron a in the hidden layer and neuron y in the output layer is calculated. When $I_s \le I_0$, the connection of a and y is disconnected. In the hidden layer, the nearest neuron b with a is found. The parameters of neuron b are adjusted as (10)–(12)

Step 6. The network output value $\hat{y} = \Phi W$ is updated. The root mean square error between the actual output and the network output e is updated. Return to Step 3

Step 7. The root mean square error e and the network output $\hat{y}$ are calculated

The trained VS-RBF neural network makes the final evaluation results closer to the evaluation results of the expert, which indicates that VS-RBF can not only replace experts to evaluate the sortie generation capacity of the carrier aircraft but also guarantee the high efficiency of evaluation. What is more, the trained VS-RBF can avoid human errors and the subjective effect on the evaluation process. At the same time, the trained VS-RBF neural network is used to establish the complex nonlinear relationship between the 15 indexes and the evaluation result. By using the nonlinear model of the indexes and the evaluation result, the influence of each index on the evaluation result can be further determined so as to provide the reference and suggestion for improving the sortie generation capacity of the carrier aircraft.

The linear transformation method is used to convert the original sample to the sample in $[0, 1]$, which is divided into two cases:

(1) When the index value is bigger, the sortie generation capacity of the carrier aircraft is better. Linear transformation is expressed in

$$Y = \frac{X - \min}{\max - \min} \tag{36}$$

(2) When the index value is smaller, the sortie generation capacity of the carrier aircraft is better. Linear transformation is expressed in

$$Y = \frac{\max - X}{\max - \min}, \tag{37}$$

where Y is the normalized sample value, X is the original sample value, and max and min are the maximum value and the minimum value of the current system, respectively

Table 7: Comparisons of average training performance.

Algorithm	Learning accuracy	Actual error	Node number of the final hidden layer	Training time (s)
VS-RBF	0.001	0.0024	31	195.38
SORBF	0.001	0.0037	42	202.42
RS-RBF	0.001	0.0041	66	418.45

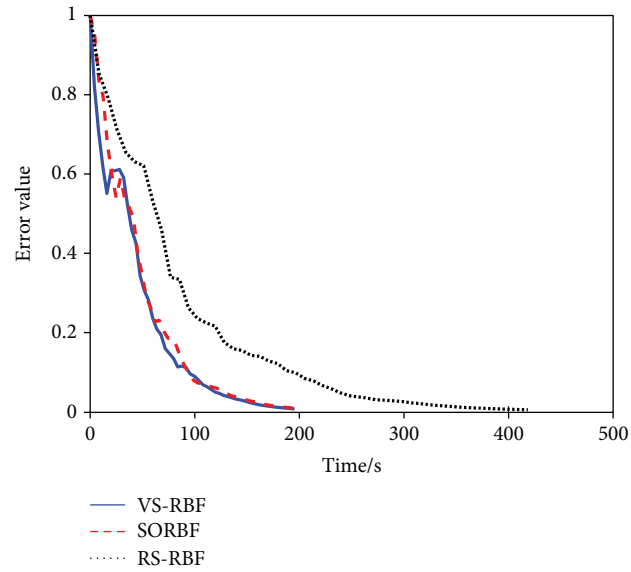


Figure 11: Error variations during training.

The initial parameters of VS-RBF, SORBF, and RS-RBF are the same. The initial weights are arbitrary values. The initial center is arbitrary value from 0 to 1. The initial function width is given to 1. The learning accuracy of the neural network is 0.001. The maximum number of iterations is 10,000.

The average performances of VS-RBF, SORBF, and RS-RBF after 50 trains are shown in Table 7. The mean square errors of VS-RBF, SORBF, and RS-RBF in the training process are shown in Figure 11. The changes of the number of neurons in the hidden layer of VS-RBF, SORBF, and RS-RBF are shown in Figure 12. The comparisons of the network output values and the actual output values of VS-RBF, SORBF, and RS-RBF in the testing process are shown in Figure 13. The errors between the network output values and the actual output values for VS-RBF, SORBF, and RS-RBF are shown in Figure 14.

The simulation shows that VS-RBF can accurately evaluate the sortie generation capacity of the carrier aircraft. Figures 13 and 14 show that evaluation value of VS-RBF agrees with the actual evaluation value. The error is less than 0.01. The evaluation error of VS-RBF is less than those of SORBF and RS-RBF. It proves the effectiveness of the evaluation of the sortie generation capacity of the carrier aircraft with VS-RBF.

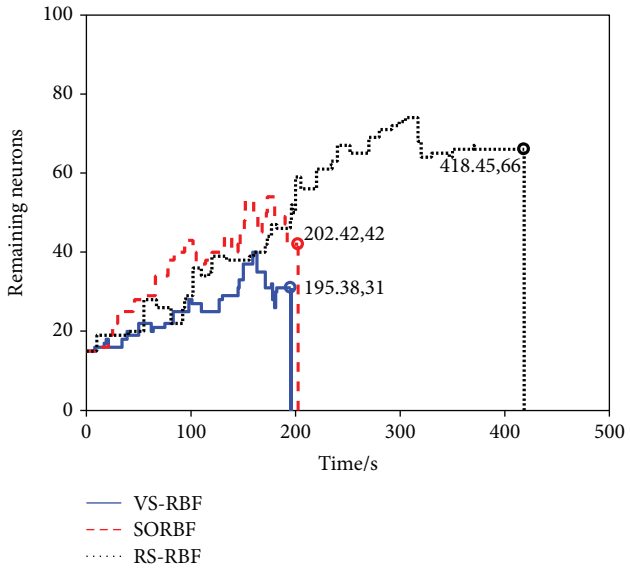


Figure 12: Number of remaining neurons during training.

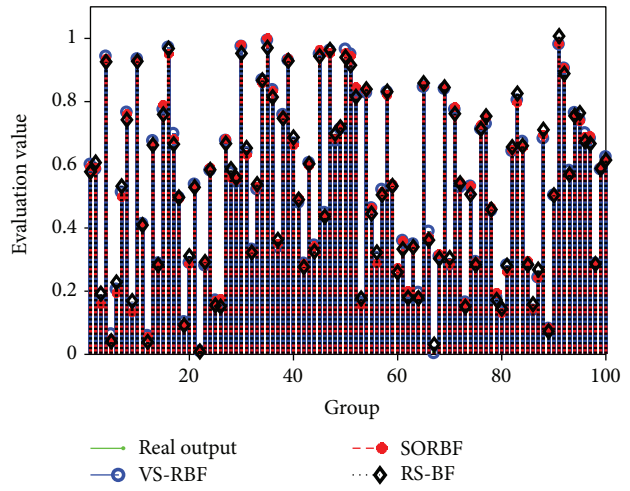


Figure 13: Output value of evaluation of the sortie generation capacity of the carrier aircraft.

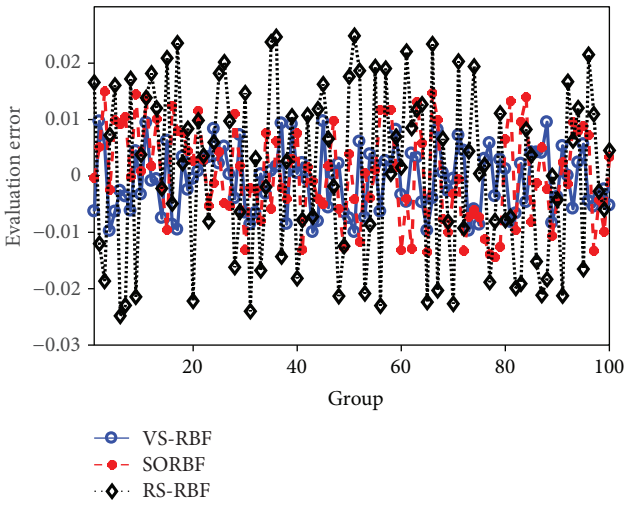


Figure 14: Evaluation error of the sortie generation capacity of the carrier aircraft.

Table 8: The reference scheme.

No.	Index	Unit	Basic configuration	Normalized result
1	ESGR	Sortie	30	0.76
2	SSGR	Sortie/day	250	0.87
3	LSGR	Sortie/day	200	0.79
4	PTP	%	80	0.91
5	MTPWP	%	11	0.64
6	MTPWR	%	9	0.58
7	SCP	%	85	0.87
8	PUR	Sortie/day	2.5	0.94
9	PIPA	%	90	0.95
10	SGRA	Sortie/day	6	0.86
11	PTNS	Minute	30	0.83
12	EI	Minute	1	0.94
13	TOOP	%	1	0.87
14	RI	Minute	1.5	0.74
15	OP	%	3.3	0.90

Figure 12 and Table 7 show that the average training time of VS-RBF is shorter than those of the other two neural network algorithms. And the final structure of the network of VS-RBF is most compact, which illustrates the effectiveness of VS-RBF in structural adjustment of neural networks. Table 7 shows that the evaluation error of VS-RBF is smallest, which shows that VS-RBF has good generalization ability.

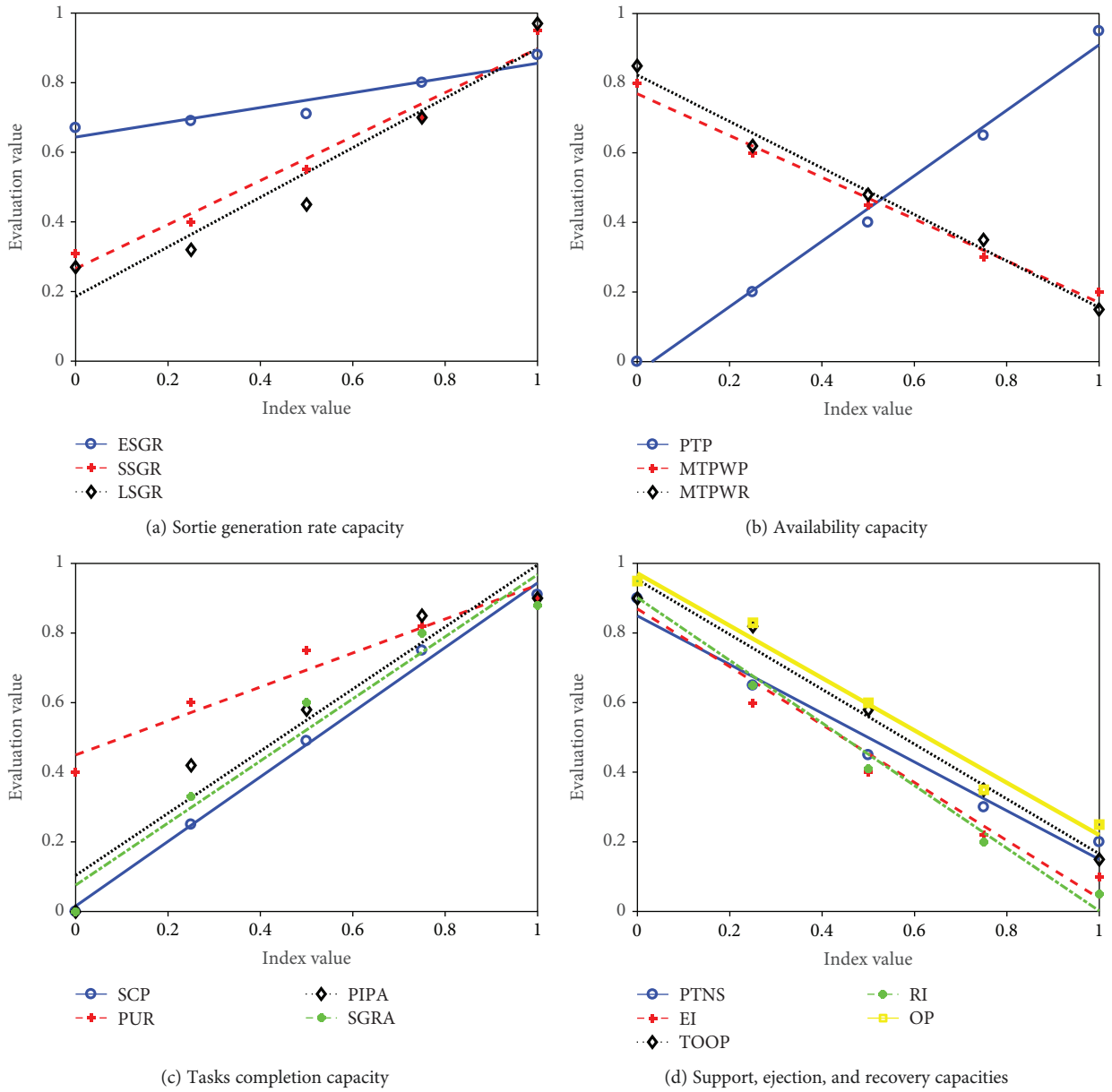
6.1. Analysis for Sortie Generation Capacity of Carrier Aircrafts. The nonlinear VS-RBF model of the indexes and evaluation result not only can be used for evaluating a given sortie generation scheme but also can further determine the effect of various indexes on the evaluation result. The scheme with basic configuration of surge operation of "Nimitz" carrier in 1997 is selected as a reference scheme, which is shown in Table 8.

On this basis, the index $x_i (i = 1, 2, \cdots, 15)$ is, respectively, adjusted as 0, 0.25, 0.5, 0.75, 1, and sortie generation capacity of the carrier aircraft is evaluated.

The 15 indexes are adjusted, respectively, and the influence curve and slope of each index on the sortie generation capacity are obtained, which are shown in Figure 15 and Table 9.

The simulations show that the VS-RBF model can be used to quickly get the influence curve of each index on the sortie generation capacity, which can be used to directly determine the factors which have a great influence on the sortie generation capacity.

Table 9 shows the fitting slope and absolute value of slope for each index. The greater the absolute value of the slope, the



(a) Sortie generation rate capacity

(b) Availability capacity

(c) Tasks completion capacity

(d) Support, ejection, and recovery capacities

Figure 15: The influence curves of 15 indexes on the sortie generation capacity.

greater the influence of the index on the sortie generation capacity. Otherwise, the smaller the absolute value of the slope, the smaller the influence of the index on the sortie generation capacity is.

Figure 15 shows that when PTP, SCP, PIPA, SGRA, or RI changes, the change of sortie generation capacity is great. So these indexes should be allocated at first in planning the scheme and guaranteed to remain the good state in the exercise or war, which will keep the sortie generation capacity high.

At the same time, Figure 15 also shows that when ESGR, SSGR, MTPWP, MTPWR, or PUR changes, the change of sortie generation capacity is small. The requirements of these indexes can be lightly reduced, which will ensure that the limited resources focus on those indexes with great influence on the sortie generation capacity and achieve the reasonable allocation of resources with low cost.

7. Conclusions

This paper proposes a VS-RBF network with the fast learning rate aimed at structuring optimization design and parameter learning algorithm of RBF. At the same time, the convergence analysis of VS-RBF is given to ensure the accuracy of the RBF neural network. By comparing with other self-organizing RBF neural networks, the following conclusions are obtained:

(1) VS-RBF can automatically adjust the structure of the RBF neural network according to the complexity of

Table 9: The slopes of 15 indexes.

No.	Index	Slope	Absolute value of slope
1	ESGR	0.2120	0.2120
2	SSGR	0.6320	0.6320
3	LSGR	0.7120	0.7120
4	PTP	0.9400	0.9400
5	MTPWP	−0.6000	0.6000
6	MTPWR	−0.6680	0.6680
7	SCP	0.9280	0.9280
8	PUR	0.4880	0.4880
9	PIPA	0.8920	0.8920
10	SGRA	0.8910	0.8910
11	PTNS	−0.7000	0.7000
12	EI	−0.8320	0.8320
13	TOOP	−0.7880	0.7880
14	RI	−0.9000	0.9000
15	OP	−0.7520	0.7520

the object. It obtains a compact RBF neural network and strong dynamic response capability

(2) The learning algorithm of parameters adapted to the structure adjustment is obtained. The fast learning rate algorithm and the robust regression algorithm improve the convergence speed of the RBF neural network

(3) The convergence analysis of the VS-RBF neural network is given. And VS-RBF has good convergence and stability

(4) Compared with several other self-organizing RBF neural networks, the proposed VS-RBF has the advantages of a compact structure, strong approximation ability, and self-organization ability. The evaluation of sortie generation capacity of the carrier aircraft with VS-RBF provides technical support for the evaluation of complex systems

To sum up, the VS-RBF proposed in this paper can effectively solve the problem of structure design and parameter learning algorithm of the RBF neural network. And the approximation of the typical nonlinear function and the evaluation of the sortie generation capacity of the carrier aircraft are realized.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] A. Jewell, M. A. Wigge, C. M. K. Gagnon, L. A. Lynn, and K. M. Kirk, *USS Nimitz and Carrier Airwing Nine Surge Demonstration*, Center for Naval Analyses, Alexandria, 1998.

[2] G. Xia, T. Luan, and M. Sun, "An evaluation method for sortie generation capacity of carrier aircrafts with principal component reduction and catastrophe progression method," *Mathematical Problems in Engineering*, vol. 2017, Article ID 2678216, 10 pages, 2017.

[3] G. Xia, T. Luan, and M. Sun, "Evaluation analysis for sortie generation of carrier aircrafts based on nonlinear fuzzy matter-element method," *Journal of Intelligent & Fuzzy Systems*, vol. 31, no. 6, pp. 3055–3066, 2016.

[4] M. H. Gilchrist, *An Evaluation of the Suitability of LCOM for Modeling. The Base-Level Munitions Production Process*, M.S. thesis, Defense Technical Information Center, Fort Belvoir, VA, USA, 1981.

[5] J. Sum and C. S. Leung, "On the error sensitivity measure for pruning RBF networks," in *Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.03EX693)*, pp. 1162–1167, Xi'an, China, November 2003.

[6] J. Platt, "A resource allocating network for function interpolation," *Neural Computation*, vol. 3, no. 2, pp. 213–225, 1991.

[7] Y. Kokkinos and K. G. Margaritis, "Topology and simulations of a Hierarchical Markovian Radial Basis Function Neural Network classifier," *Information Sciences*, vol. 294, pp. 612–627, 2015.

[8] R. Xia, J. Huang, Y. Chen, and Y. Feng, "A study of the method of the thermal conductivity measurement for VIPs with improved RBF neural networks," *Measurement*, vol. 87, no. 5, pp. 246–254, 2016.

[9] M. Aghbashlo, S. Hosseinpour, M. Tabatabaei, A. Dadak, H. Younesi, and G. Najafpour, "Multi-objective exergetic optimization of continuous photo-biohydrogen production process using a novel hybrid fuzzy clustering-ranking approach coupled with radial basis function (RBF) neural network," *International Journal of Hydrogen Energy*, vol. 41, no. 41, pp. 18418–18430, 2016.

[10] M. K. Kadalbajoo, A. Kumar, and L. P. Tripathi, "A radial basis function based implicit–explicit method for option pricing under jump-diffusion models," *Applied Numerical Mathematics*, vol. 110, pp. 159–173, 2016.

[11] N. Zhang, S. Ding, and J. Zhang, "Multi layer ELM-RBF for multi-label learning," *Applied Soft Computing*, vol. 43, no. 9, pp. 535–545, 2016.

[12] E. Assareh, I. Poultangari, E. Tandis, and M. Nedaei, "Optimizing the wind power generation in low wind speed areas using an advanced hybrid RBF neural network coupled with the HGA-GSA optimization method," *Journal of Mechanical Science and Technology*, vol. 30, no. 10, pp. 4735–4745, 2016.

[13] M. Dehghan and M. Safarpoor, "The dual reciprocity boundary elements method for the linear and nonlinear two-dimensional time-fractional partial differential equations," *Mathematical Methods in the Applied Sciences*, vol. 39, no. 14, pp. 3979–3995, 2016.

[14] M. Dehghan and R. Salehi, "A method based on meshless approach for the numerical solution of the two-space dimensional hyperbolic telegraph equation," *Mathematical Methods in the Applied Sciences*, vol. 35, no. 10, pp. 1220–1233, 2012.

[15] Z. G. Zhu and S. L. Tian, "Improvement of learning rate of feed forward neural network based on weight gradient," *Computer Systems & Applications*, vol. 27, no. 7, pp. 205–210, 2018.

[16] J. Wang, C. Zhang, H. Zhu, X. Huang, and L. Zhang, "RBF nonsmooth control method for vibration of building structure with actuator failure," *Complexity*, vol. 2017, Article ID 2513815, 7 pages, 2017.

[17] W. Liu, Z. Yang, and K. Bi, "Forecasting the acquisition of university spin-outs: an RBF neural network approach," *Complexity*, vol. 2017, Article ID 6920904, 8 pages, 2017.

[18] Y. Yin, H. Niu, and X. Liu, "Adaptive neural network sliding mode control for quad tilt rotor aircraft," *Complexity*, vol. 2017, Article ID 7104708, 13 pages, 2017.

[19] L. Wang, Y. Cheng, J. Hu, J. Liang, and A. M. Dobaie, "Nonlinear system identification using quasi-ARX RBFN models with a parameter-classified scheme," *Complexity*, vol. 2017, Article ID 8197602, 12 pages, 2017.

[20] H. Li and W. Sun, "Study on BP neural network learning efficiency," *Value Engineering*, vol. 37, no. 12, pp. 183–186, 2018.

[21] J. Xie, F. U. Bing, and J. Bao, "Analysis of the sortie generation capacity of embarked airwings by using the state transition diagram," *Chinese Journal of Ship Research*, vol. 9, no. 2, pp. 1–5, 2014.

[22] X. C. Liu, J. Lu, and X. Z. Huang, "Analysis on the index system of sortie generation capacity of embarked aircrafts," *Chinese Journal of Ship Research*, vol. 6, no. 4, pp. 1–7, 2011.

[23] X. G. Zhou, R. H. Zhao, S. Y. Wang, and J. He, "Research on influence of flight deck operation exerts upon sortie generation of carrier-based aircraft," *Journal of System Simulation*, vol. 26, no. 10, pp. 2447–2451, 2014.

[24] W. Wang and S. W. Yan, "Analysis on evaluation method of SGR of US aircraft carrier," *Ship Science and Technology*, vol. 38, no. 4, pp. 140–142, 2016.

[25] Y. L. Zhang, R. K. Zhou, W. P. Ji, and C. B. Fu, "Efficiency evaluation for carrier formation swarming aircraft based on fuzzy synthetic evaluation method," *Ship Electronic Engineering*, vol. 35, no. 10, pp. 117–121, 2015.

[26] X. Y. Yin, Q. Zou, and J. C. Feng, "Evaluation analysis for antiship combat capability of carrier aircraft inertial," *Fire Control & Command Control*, vol. 42, no. 12, pp. 40–44, 2017.

[27] Y. L. Zhang, Z. S. Sun, C. H. Chen, and Y. H. Hu, "Research into the threat evaluation of carrier-borne aircraft and cruise missile," *Shipboard Electronic Contermeasure*, vol. 40, no. 2, pp. 1–5, 2017.

[28] Y. B. Ding, D. D. You, and C. F. Guo, "Quantitative evaluating model of effective carrier-based aircraft in air defence of the carrier formation," *Journal of Ordnance Equipment Engineering*, vol. 38, no. 7, pp. 46–49, 2017.

[29] H. Zheng, C. Zhang, Y. Wang, J. Sladek, and V. Sladek, "A meshfree local RBF collocation method for anti-plane transverse elastic wave propagation analysis in 2D phononic crystals," *Journal of Computational Physics*, vol. 305, no. 5, pp. 997–1014, 2016.

[30] H. Sarimveis, A. Alexandridis, S. Mazarakis, and G. Bafas, "A new algorithm for developing dynamic radial basis function neural network models based on genetic algorithms," *Computers and Chemical Engineering*, vol. 28, no. 1-2, pp. 209–217, 2004.

[31] H. G. Han, Q. L. Chen, and J. F. Qiao, "An efficient self-organizing RBF neural network for water quality prediction," *Neural Networks*, vol. 24, no. 7, pp. 717–725, 2011.

[32] Z. Q. Wu, W. J. Jia, L. R. Zhao, and C. H. Wu, "Maximum wind power tracking based on cloud RBF neural network," *Renewable Energy*, vol. 86, no. 10, pp. 466–472, 2016.

[33] X. Fu and L. Wang, "Data dimensionality reduction with application to simplifying RBF network structure and improving classification performance," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 33, no. 3, pp. 399–409, 2003.

[34] J.-X. Peng, K. Li, and d.-S. Huang, "A hybrid forward algorithm for RBF neural network construction," *IEEE Transactions on Neural Networks*, vol. 17, no. 6, pp. 1439–1451, 2006.

[35] H. Han, Q. Chen, and J. Qiao, "Research on an online self-organizing radial basis function neural network," *Neural Computing and Applications*, vol. 19, no. 5, pp. 667–676, 2010.

[36] S. Ding, G. Ma, and Z. Shi, "A rough RBF neural network based on weighted regularized extreme learning machine," *Neural Processing Letters*, vol. 40, no. 3, pp. 245–260, 2014.

[37] Y. Liu, J. A. Starzyk, and Z. Zhu, "Optimized approximation algorithm in neural networks without overfitting," *IEEE Transactions on Neural Networks*, vol. 19, no. 6, pp. 983–995, 2008.