

Research Article

A Novel Nonlinear Function Fitting Model Based on FOA and GRNN

Yongli Zhang ¹, Jianguang Niu ¹ and Sanggyun Na²

¹School of Management Science and Engineering, Hebei GEO University, Shijiazhuang, Hebei Province, China

²College of Business Administration, Wonkwang University, Iksan, Jeonbuk, Republic of Korea

Correspondence should be addressed to Yongli Zhang; zhangyongli086@163.com

Received 31 August 2018; Revised 24 December 2018; Accepted 16 January 2019; Published 5 February 2019

Academic Editor: Erik Cuevas

Copyright © 2019 Yongli Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The nonlinear function fitting is an essential research issue. At present, the main function fitting methods are statistical methods and artificial neural network, but statistical methods have many inherent strict limits in application, and the back propagation (BP) neural network used widely has too many optimized parameters. For the gaps and lacks of existing researches, the FOA-GRNN was proposed and compared with the GRNN, GA-BP, PSO-BP, and BP through three nonlinear functions from simplicity to complexity for verifying the accuracy and robustness of the FOA-GRNN. The experiment results showed that the FOA-GRNN had the best fitting precision and fastest convergence speed; meanwhile the predictions were stable and reliable in the Mexican Hat function and Rastrgrin function. In the most complex Griewank function, the prediction of FOA-GRNN was becoming unstable and the model did not show better than GRNN model adopting equal step length searching method, but the performance of FOA-GRNN is superior to that of GA-BP, PSO-BP, and BP. The paper presents a new approach to optimize the parameter of GRNN and also provides a new nonlinear function fitting method, which has better fitting precision, faster calculation speed, more few adjusted parameters, and more powerful processing ability for small samples. The processing capacity of FOA for treating high complex nonlinear function needs to be further improved and developed in the future study.

1. Introduction

In various fields of science and technology, the relationship between different variables is usually described by the function. Some functional relations can be derived from classical theoretical analysis, which not only provides theoretical basis for further analysis and research, but also facilitates the solution of practical engineering problems. However, many engineering problems are difficult to directly derive the functional expression between variables, or even if the expression can be obtained, the formula is very complex, which is not conducive to further analysis and calculation. Due to the research needs, it is expected to obtain the functional relationship between these variables. At this time, the fitting method can be used to obtain the approximate functional expression between variables by combining the known experimental data with the mathematical method.

According to the fitting problem, the fitting methods are generally divided into linear fitting methods and nonlinear

fitting methods. For the linear data fitting problem, linear fitting methods usually adopt a set of simple, linearly independent basis functions to approximate the experimental data. However, the linear fitting methods have many inherent strict limits in practical application; assume that independent variables are normal distribution, nonrelevance, and independent; function form relating to independent variables and dependent variable must be preexisting and linearity [1]. For the nonlinear data fitting problem, it is usually divided into two cases for processing. One way is to use variable substitution to turn it into a linear problem and then solve it. The other is the problem that cannot linearize (processing is more troublesome) and can use the method such as artificial neural network to achieve. The artificial neural network has strong mapping and generalization ability, which can self-organize, self-study, and fit the arbitrary nonlinear relationship between dependent variable and independent variables without accurate mathematical model [2, 3]. The back propagation (BP) neural network is one of most popular

artificial neural networks, but the BP neural network needs a large number of data samples for training and also has too many optimized parameters.

Generalized regression neural network (GRNN) has excellent performance of anti-interference, good approximation ability, fast convergence speed, and the ability of autonomous learning. In addition, in the treatment of small sample or unstable data, the GRNN has a better effect [4, 5]. Currently, GRNN is widely used in engineering fields [6–10]. This study contributes to existing studies by applying GRNN for nonlinear function approximation and comparing with linear regression model and BP neural network to assess the strengths and weaknesses of GRNN.

Although the GRNN does not need preexisting function model, the SPREAD parameter of GRNN has significant influence on the prediction performance. Therefore, in order to select an appropriate SPREAD parameter, researchers have raised multiple evolutionary algorithms for optimizing neural network, such as genetic algorithm (GA) [11, 12], particle swarm optimization (PSO) [13, 14], tabu search algorithm (TSA) [15, 16], ant colony algorithm (ACA) [17, 18], etc.

However, traditional evolutionary algorithms have common disadvantages, such as complex program, difficulty to understand, slow convergence rate, etc., so the scholar Pan proposed the fruit fly optimization algorithm (FOA) in 2012 [19]. Currently, few studies have conducted FOA on the optimization of GRNN. Although FOA-GRNN model was further designed by Yongli Zhang et al. in 2018, FOA-GRNN was only directly applied to the research issue of technological innovation, not compared with other algorithms or models to verify its advantages in the algorithm performance such as nonlinear fitting ability and predictive stability [20]. Therefore, this study is undertaken to assess various intelligent algorithms to test whether FOA is the accurate optimization algorithm with best fitting precision and fastest calculation speed, to fill the research gap.

This paper utilized the fruit fly optimization algorithm (FOA) to optimize the SPREAD parameter of GRNN and established the FOA-GRNN model and then designed three nonlinear functions from simplicity to complexity, compared FOA-GRNN with GRNN, GA-BP, PSO-BP, and BP models in the same experimental conditions for verifying the accuracy and robustness of the FOA-GRNN model.

The rest of this study is organized as follows. The principle of GRNN, FOA, GA, PSO, and BP are described in Section 2. The simulation experiments are given in Section 3. Finally, conclusions are drawn and future research is suggested in Section 4.

2. Generalized Regression Neural Network

2.1. The Basic Theory of GRNN. In 1991, American scholar Donald F. Specht proposed the generalized regression neural network (GRNN), which is a special form of radial basis function (RBF) neural network [21]. GRNN has strong nonlinear mapping ability and is suitable for solving various nonlinear problems. In addition, when the sample data

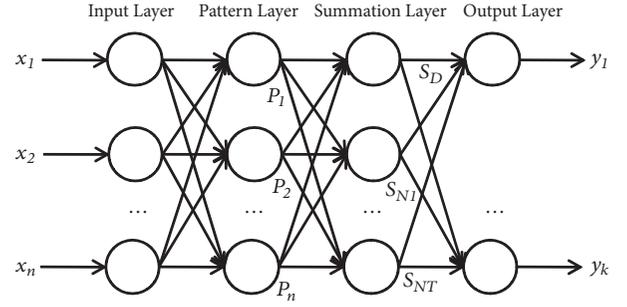


FIGURE 1: The topological structure of GRNN.

is small, GRNN also has a better prediction effect. At present, GRNN has been widely applied in various fields, such as enterprise operation management, bioengineering, energy demand forecasting, structural analysis, and control decision-making system [5, 10, 22–24].

The network structure of GRNN is composed of input layer, pattern layer, summation layer, and output layer, as shown as Figure 1, wherein $X = [x_1, x_2, \dots, x_n]^T$ represents network input vector and $Y = [y_1, y_2, \dots, y_k]^T$ represents network out vector.

The number of neurons in the input layer is equal to the dimension n of the input vector in the learning sample. Each neuron is a simple distribution unit, and these neurons directly pass input variables to the pattern layer.

The number of neurons in the pattern layer is the number n of learning samples, and each neuron corresponds to one learning sample. The transfer function of the i^{th} neuron in the model layer is as follows.

$$P_i = \exp \left[-\frac{(X - X_i)^T (X - X_i)}{2\sigma^2} \right], \quad i = 1, 2, \dots, n \quad (1)$$

In formula (1), P_i is the output of the i^{th} neuron in the pattern layer, σ is the smoothing factor, and X_i is the learning sample corresponding to the i^{th} neuron.

The summation layer contains the summation unit in the denominator and the summation unit in the numerator. The summation unit in the denominator performs arithmetic summation on the outputs of all neurons in the pattern layer. The sum of connection weights between all neurons of pattern layer and a neuron in the summation layer are 1, and the transfer function is shown in formula (2).

$$S_D = \sum_{i=1}^n P_i \quad (2)$$

The molecular summation unit is a weighted summation of the output of neurons in the pattern layer; the element j in the output sample Y_j is the connection weight between pattern layer neuron i and summation layer neuron j , as shown in formula (3).

$$S_{Nj} = \sum_{i=1}^n y_{ij} P_i, \quad j = 1, 2, \dots, k \quad (3)$$

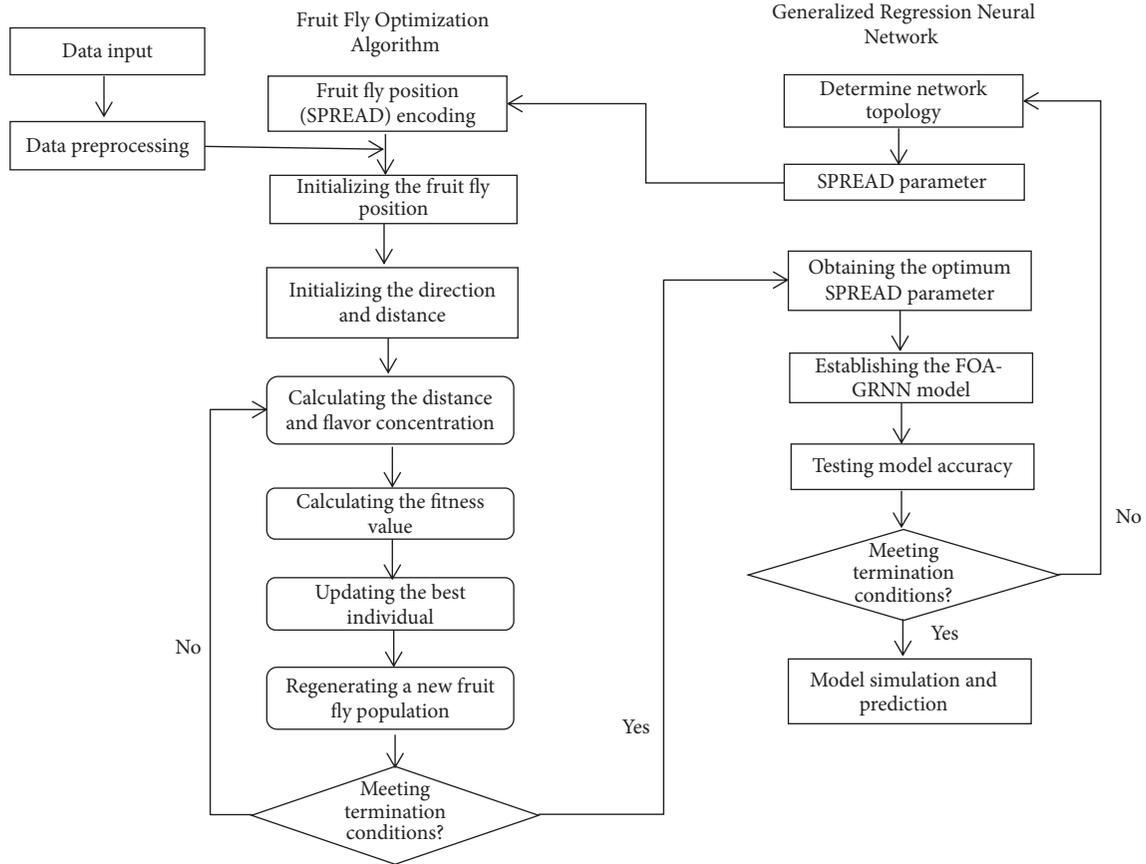


FIGURE 2: The parameter optimization process of FOA-GRNN model.

The output layer Y is divided by the output of the numerator summation unit and the denominator summation unit in the summation layer, as shown in formula (4).

$$Y_j = \frac{S_{Nj}}{S_D}, \quad j = 1, 2, \dots, k \quad (4)$$

2.2. FOA-GRNN Model. When using GRNN for modeling, as long as the training sample is determined, the corresponding network structure and the connection weight between neurons are also determined accordingly, so that the stability of GRNN is very good; its training process is actually the process of optimizing smooth parameter σ . By changing the parameter σ , the transfer function of the hidden layer is adjusted to obtain the best regression estimation result. If the value of parameter σ is not appropriate, GRNN cannot achieve the expected training effect. Because the simulation tool used in this paper is MATLAB, the parameter SPREAD is used to express the smooth parameter σ .

In order to obtain the ideal parameter SPREAD and make the prediction model have good generalization ability, fruit fly optimization (FOA) algorithm is adopted to optimize the smooth parameter SPREAD in GRNN model.

Fruit fly optimization algorithm (FOA) is a global optimization algorithm by simulating fruit fly foraging behavior.

FOA has the advantages of simple structure, low computation complexity, and high execution efficiency, proposed by scholar Pan Wen-Chao in 2012 [19]. The optimization process for searching the parameter SPREAD of GRNN model through FOA is shown in Figure 2.

The specific optimization steps of FOA-GRNN are as follows.

Step 1. Randomly initialize fruit fly position, setting iteration number, and population size.

Step 2. Randomly set the direction and distance of each fruit fly.

Step 3. Calculate the distance between the fruit fly and the origin ($D_i^2 = X_i^2 + Y_i^2$) and the determination value of flavor concentration ($S=1/D_i$).

Step 4. Establish the fitness function. One flavor concentration determination value S represents one position of fruit fly, then it is regarded as one SPREAD parameter of GRNN, substituted into the fitness function to obtain the fitness of the position of fruit fly. In this study, the reciprocal of root mean square error (RMSE) between the predicted values and actual values is treated as the fitness function of fruit fly.

Step 5. Regenerate a new fruit fly population. The best fruit fly with highest fitness is pick out, then around the best fruit fly, the new fruit flies will be regenerated and born. The direction and distance of new fruit fly is adjusted based on the position of the best fruit fly of previous generation.

Step 6. Repeat Steps 3 to 5 and the fruit fly iteratively searches for the optimal parameter to determine whether the flavor concentration is better than the previous iteration.

Step 7. Judge whether to meet the requirements of the terminating condition, finally, obtaining the optimal parameter SPREAD and GRNN model.

2.3. Other Intelligent Algorithms

2.3.1. BP Neural Network. Back propagation (BP) neural network is a multilayer feed forward network proposed by Rumelhart and McClelland in 1986, and it is one of neural network modes applied widely. It can self-learn, self-organize, and fit any nonlinear function. During training process, the BP neural network takes minimum sum of squared errors as learning goal and continuously adjusts the weights and thresholds via steepest descent method to approach the desired output [25]. Due to learning rule is steepest descent method, the BP neural network easily falls into local optima, unable to obtain the global optimum [26].

2.3.2. Genetic Algorithm. Genetic algorithm (GA) is an evolutionary algorithm proposed by American professor Holland in 1962. It follows along with the biology inheritance and the evolutionary mechanism of “survival of the fittest” to search the global optimal solution. During optimum process, genetic algorithm firstly transforms the problem solutions into population individuals embodied by code and initializes population randomly, in which each individual represents one candidate solution. Then genetic algorithm calculates the fitness of each population individual, reserves superior individuals with high fitness, and eliminates inferior individuals with low fitness through selection, crossover, and mutation operations; evolution repeats until the best individual is searched, which is decoded to obtain the optimal solution [20, 26].

2.3.3. Particle Swarm Optimization Algorithm. Particle swarm optimization (PSO) is an evolutionary computation technique based on swarm intelligence. It stems from the simulation of bird flock’s looking for food and has been proposed by Kennedy and Eberhart in 1995. The PSO algorithm maps the solution space to the particle swarm, and each particle is characterized with fitness, position, and speed. The fitness distinguishes superior and inferior particle; the speed determines the direction and distance of particle moving. The optimal position particle pass is called “private best position”, and the optimal position in entire particle swarm’s best position is called “global best position”. When particle moves in search space, each particle’s movement is guided by private best position as well as global best position.

The better particle will be found around the private best position and global best position, which will be updated until obtaining the optimal solution represented by global best position [13, 27].

Weights and thresholds of BP neural network have great influence on the fitting precision. Therefore, in order to continuously enhance fitting precision, genetic algorithm and particle swarm optimization algorithm are introduced to BP neural network for obtaining the optimal weights and thresholds, and finally, the BP neural network optimized by genetic algorithm (GA-BP) and BP neural network optimized by particle swarm optimization algorithm (PSO-BP) are established.

3. Simulation Experiments

In the sequence from simplicity to complexity, we designed three nonlinear functions. Each prediction model is set to the same experimental conditions. Using each nonlinear function, our study randomly generated 10,000 data, and the first 8000 were considered as training data; the following 2000 were treated as testing data.

3.1. Parameters Setting and Performance Criteria. For verifying the accuracy and robustness of the FOA-GRNN model, the FOA-GRNN is compared with GRNN, GA-BP, PSO-BP, and BP. The experimental conditions are set to equal, that means input and output variables, training and testing dataset, fitness function, population scale, and loops number are the same.

In FOA-GRNN, the SPREAD parameter of GRNN is the optimal targets of FOA; one fruit fly position represents one parameter. In GA-BP and PSO-BP, the weights and thresholds of BP are regarded as the population individual of GA and PSO. The reciprocal of root mean square error (RMSE) between prediction values and actual values is treated as individual fitness function. Population size and evolution generations of each evolutionary algorithm are set to 20 and 20.

Moreover, crossover and mutation probability of genetic algorithm is set to 0.2 and 0.1. In PSO, the particle maximum, particle maximum, velocity maximum, velocity minimum, and acceleration constants $c1$ & $c2$ are set to 0.55, 0.05, 1, -1, and 1.49445. For BP neural network, the network topological structure is set to 2-4-1, the convergence criterion is RMSE less than or equal to 0.0001, and the iteration maximum, learning rate, and training function are 100, 0.1, and traingd (Table 1).

The root mean square error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE) are used to assess the fitting precision of models. The smaller the value of RMSE, MAE, or MAPE is, the less the forecasting error is, the better the model fit is, and vice versa.

The formulae of three performance criteria are shown as follows:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2} \quad (5)$$

TABLE 1: Parameters setting of algorithms.

Model Type	Training and Testing samples	Algorithm parameters
FOA-GRNN	Training samples: 8000 dataset Testing samples: 2000 dataset	Individual: the SPREAD parameter; Individual fitness function: Sum of absolute errors; Evolution generations:20; Population size:20;
GA-BP	Same	Individual: the weights and thresholds of BP network; Individual fitness function: Same; Evolution generations: Same; Population size: Same; Crossover probability: 0.2; Mutation probability: 0.1;
PSO-BP	Same	Individual: the weights and thresholds of BP network; Individual fitness function: Same; Evolution generations: Same; Population size: Same; Particle maximum:0.55; Particle minimum:0.05; Velocity maximum:1; Velocity minimum: -1; Acceleration constants c1& c2: 1.49445;
GRNN	Same	Individual: Same; Individual fitness function: Same; Evolution generations: Same; Population size: Same;
BP	Same	Individual: the weights and thresholds of BP network; Individual fitness function: Same; Evolution generations: Same; Population size: Same;

TABLE 2: The predictive comparison of first testing function.

Model	Total Error	RMSE	MAE	MAPE	Stability
FOA-GRNN	57.2698	0.0412	0.0286	0.1468	Yes
GRNN	138.3150	0.0961	0.0692	0.3719	Yes
GA-BP	833.1369	0.5256	0.4166	2.0451	No
PSO-BP	768.0228	0.4824	0.3840	1.7451	No
BP	861.4234	0.5436	0.4307	2.3267	No

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{y}_i - y_i| \quad (6)$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{\hat{y}_i - y_i}{y_i} \right| \quad (7)$$

where y_i represents actual output value, \hat{y}_i represents predicted output value, and n is sample size.

3.2. *Experiment 1: Mexican Hat Function.* The first nonlinear function for testing is Mexican Hat function [28, 29], shown in formula (8) and Figure 3.

$$f(x, y) = \frac{\sin \sqrt{x^2 + y^2}}{\sqrt{x^2 + y^2}} + e^{(\cos 2\pi x + \cos 2\pi y)/2} - 2.71289 \quad (8)$$

In formula (8), the range of the variables x and y is $[-2, 2]$.

To search for the best population individual, FOA-GRNN, GRNN, GA-BP, and PSO-BP evolved 20 generations; the fitness value of best population individual in every generation appeared in Figure 4.

As shown in Figure 4, FOA-GRNN and PSO-BP begin to converge at the 4 and 16 generations, GRNN adopting equal step length searching method cannot capture the optimal speed parameter, and the GA-BP model converges slowly. The comparison result is shown in Table 2.

In Table 2, the most correct predictions are exhibited by FOA-GRNN (Total Error=57.2698, RMSE=0.0412, MAE=0.0286, and MAPE=0.1468), followed by GRNN (Total Error=138.3150, RMSE=0.0961, MAE=0.0692, and MAPE=0.3719). Due to various local extrema of testing function, the GA-BP (Total Error=833.1369, RMSE=0.5256, MAE=0.4166, and MAPE=2.0451), PSO-BP (Total Error=768.0228, RMSE=0.4824, MAE=0.3840, and MAPE=1.7451) models and BP (Total Error=861.4234, RMSE=0.5436, MAE=0.4307, and MAPE=2.3267) perform poorly. Figure 4

TABLE 3: The predictive comparison of second testing function.

Model	Total Error	RMSE	MAE	MAPE	Stability
FOA-GRNN	2865.8925	1.9038	1.4329	0.0514	Yes
GRNN	3252.8647	2.0885	1.6264	0.0613	Yes
GA-BP	16520.2570	10.1086	8.2601	0.3346	No
PSO-BP	16482.0882	10.1015	8.2410	0.3373	No
BP	16505.6928	10.1119	8.2528	0.3362	No

TABLE 4: The predictive comparison of testing function 3.

Model	Total Error	RMSE	MAE	MAPE	Stability
FOA-GRNN	686.0692	0.4455	0.3430	0.1971	No
GRNN	685.7656	0.4455	0.3429	0.1969	Yes
GA-BP	823.8674	0.5083	0.4119	0.2415	No
PSO-BP	822.9429	0.5080	0.4115	0.2431	No
BP	825.5087	0.5090	0.4128	0.2430	No

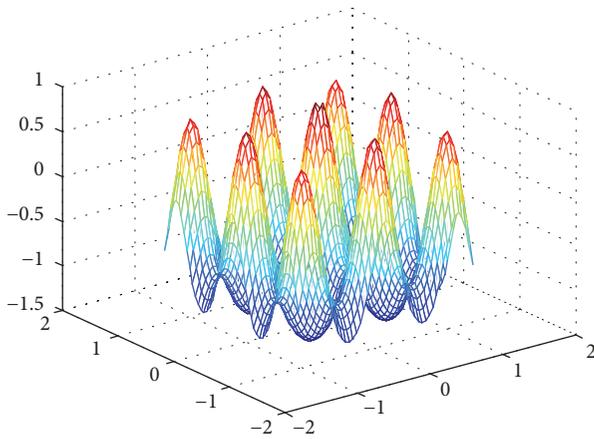


FIGURE 3: The function graph of first nonlinear function.

and Table 2 proved that FOA-GRNN had the fastest convergence speed and best fitting precision.

Otherwise, with the same training and testing data, the programs of FOA-GRNN, GRNN, GA-BP, PSO-BP, and BP models had been run repeatedly for several times, each running of FOA-GRNN and GRNN models returned the same result, but the predicted results of GA-BP, PSO-BP, and BP models were differently within the same test scenario. That indicated the FOA-GRNN and GRNN models had been trained sufficiently and had gained the optimal solution, the predictions of FOA-GRNN and GRNN models were stable and reliable. However, for GA-BP, PSO-BP, and BP models, the sample size was small; thorough training had not been received, so the optimal solution could not be obtained; models exhibited unsaturated and unstable state

3.3. Experiment 2: Rastrgrin Function. The second nonlinear function for testing is Rastrgrin function [30, 31]; its formula and graph are shown in formula (9) and Figure 5.

$$f(x, y) = x^2 + y^2 - 10 \times \cos 2\pi x - 10 \times \cos 2\pi y + 20 \quad (9)$$

In formula (9), the range of the variables x and y is $[-5, 5]$.

Figure 6 revealed that FOA-GRNN had the fastest convergence speed and the highest accuracy, the GRNN adopting equal step length searching method cannot capture the optimal speed parameter. The detailed predictive comparison is shown in Table 3.

Table 3 certified that the most accurate prediction is FOA-GRNN, followed by GRNN, PSO-BP, GA-BP, and BP. Meanwhile, the FOA-GRNN and GRNN models were stable; PSO-BP, GA-BP, and BP were unsaturated and unstable.

3.4. Experiment 3: Griewank Function. The third nonlinear function for testing is Griewank function [32, 33]. It is described in formula (10) and Figure 7.

$$f(x, y) = \frac{x^2 + y^2}{4000} - \cos x \times \cos \frac{y}{\sqrt{2}} + 1 \quad (10)$$

In formula (10), the range of the variables x and y is $[-100, 100]$.

The third testing function is more complex, with more local extreme values. Figure 8 and Table 4 revealed that, in convergence speed and fitting precision, FOA-GRNN is superior to GA-BP and PSO-BP, but it has no advantage over GRNN adopting equal step length searching method; the FOA needs further improvement and development.

4. Conclusion and Discussion

In response to the gaps and shortages of current prediction methods, this study established FOA-GRNN model, compared FOA-GRNN model with other four prediction models raised by previous scholar (GRNN, GA-BP, PSO-BP, and BP). Simulation results showed that the FOA-GRNN model proposed by this study had advantageous properties such as better fitting precision, faster calculation speed, fewer adjusted parameters, and stronger nonlinear fitting capability. The main conclusions are summarized as follows.

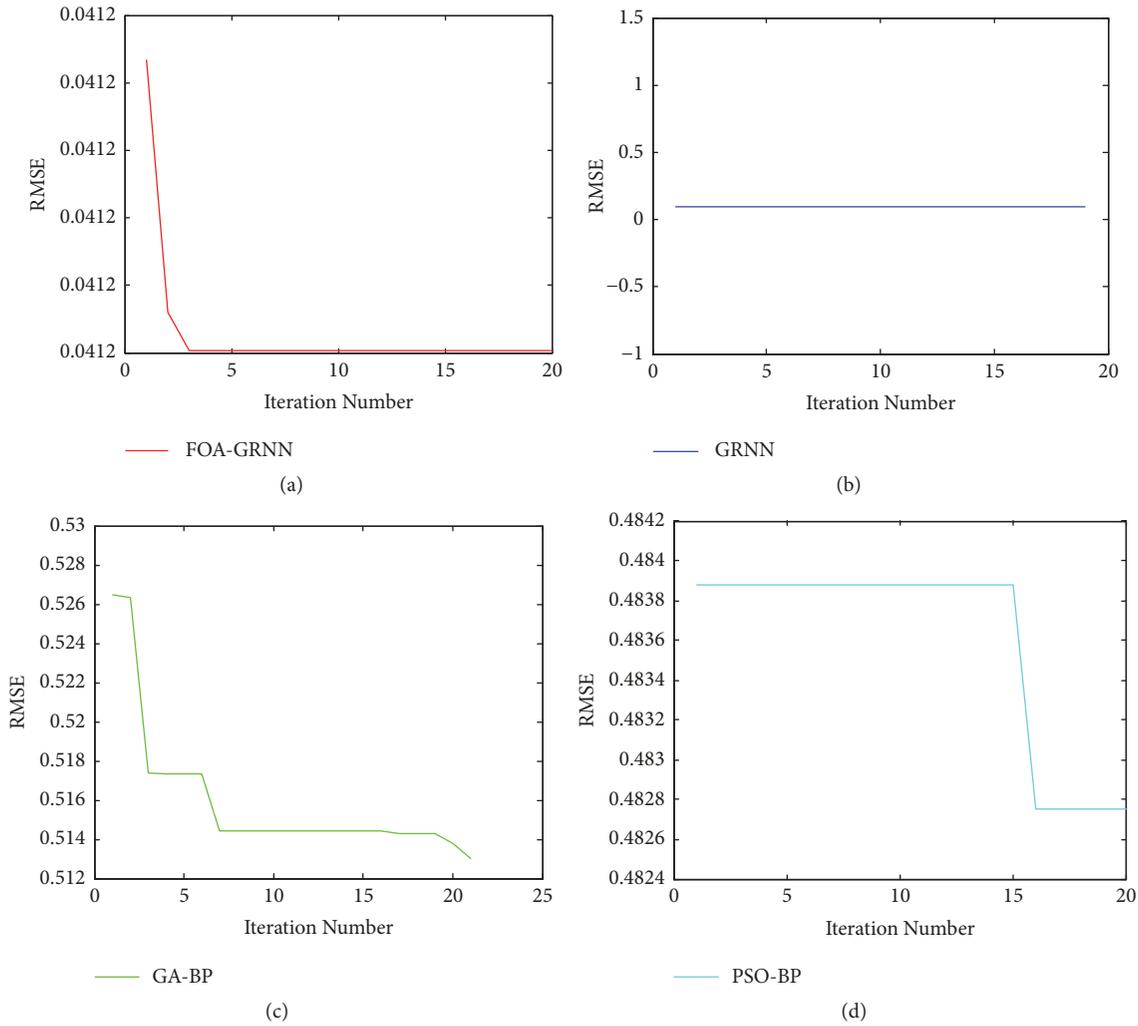


FIGURE 4: The optimization process of first testing function.

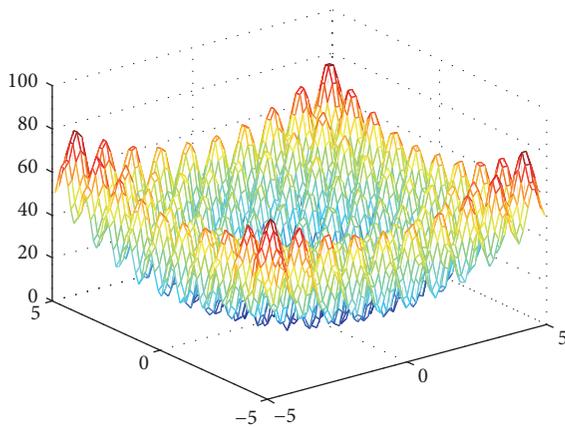


FIGURE 5: The function graph of second nonlinear function.

First, compare with BP model, GRNN model has fewer adjusted parameters, better fitting precision, and more suitable for small samples. The parameters searched by BP

neural network model are the whole network weights and thresholds, so the number of parameters is huge. For example, in this study, BP neural network has 2 input nodes, 2 hidden nodes, and 1 output node; the parameters required to be sought are 17. Therefore, the BP neural network model is difficult to obtain global optimization and easily fall into the local extremum, and the convergence rate is relatively slow. Moreover, BP needs larger training dataset; the small or incomplete sample data will lead to insufficient training and uncertain predictions. But the parameter needed to be adjusted by GRNN model is only one, so the GRNN model has better fitting precision, faster convergence speed, and more suitable for small sample data.

Second, FOA can get global optimum solution more rapidly and greatly enhance the convergence rate of the GRNN model. FOA which is an artificial intelligence algorithm is put forward by foraging act of research and observation about groups of fruit flies. Compared with random assignment or loops with step-in for searching parameters, the FOA can obtain the best SPREAD parameter faster and improve the convergence speed of the GRNN model.

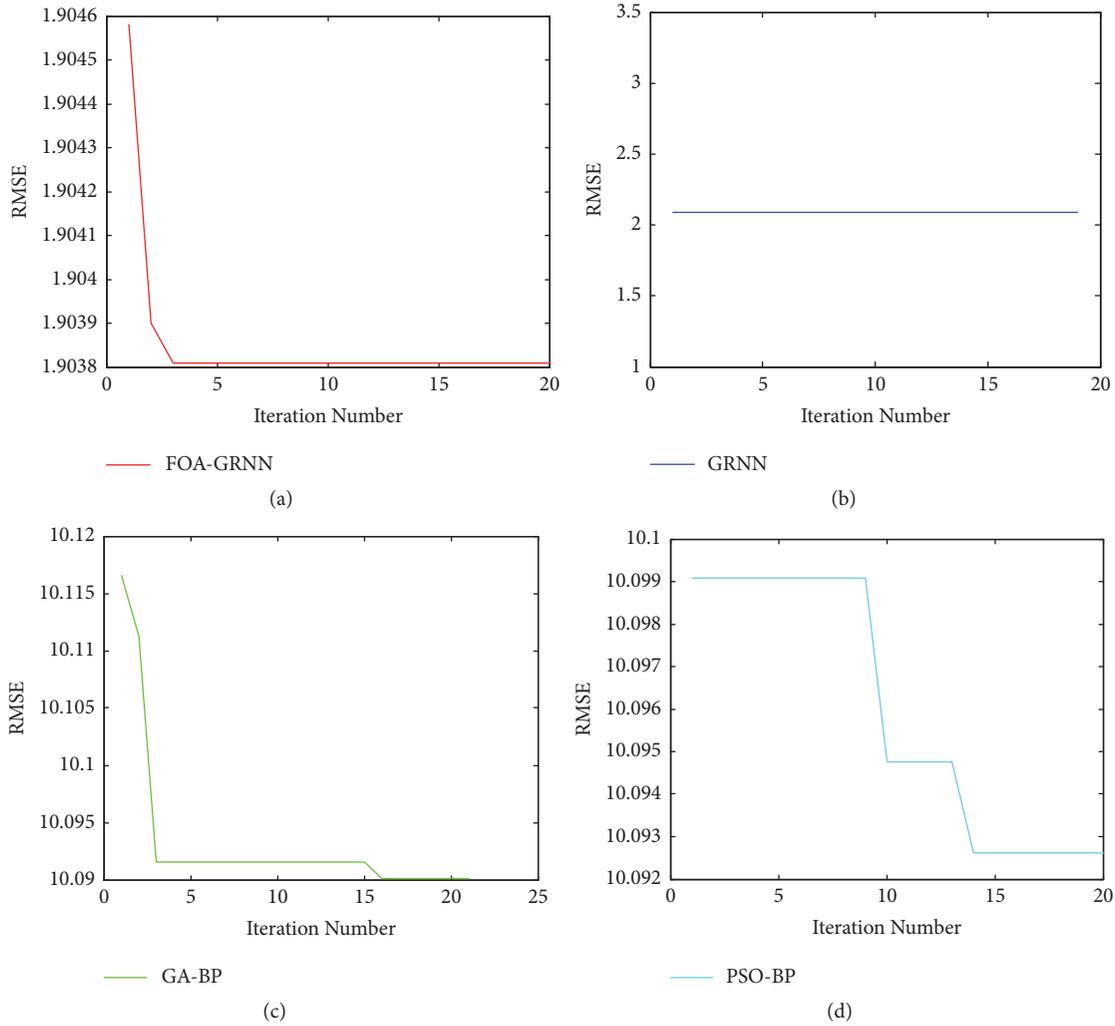


FIGURE 6: The optimization process of second testing function.

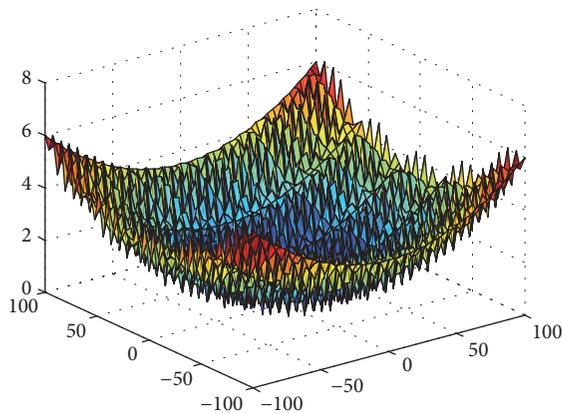


FIGURE 7: The function graph of third nonlinear function.

Third, the fitting precision of optimized BP neural network model is better than that of BP neural network model. In GA or PSO, population individuals are filtered according

to fitness value, superior individuals with high fitness will be retained, and inferior individuals with low fitness will be eliminated; the new generation inherits the previous generation, and also superior to the previous generation, so cycle evolution, until meeting the termination conditions, gets the fittest individual. The GA or PSO algorithm obtains the optimal weights and thresholds of BP neural network more easy and quick than the random search method. Therefore, the predictions of GA-BP and PSO-GA are more accurate than BP neural network model.

Nevertheless, more research is needed to present opportunities for making perfect predictions. As the nonlinear function becomes more and more complex, the local extrema are increasing, and the improvement effect of FOA is getting smaller and smaller, which indicates that the processing capacity for treating complex nonlinear function of FOA needs to be further improved and developed. The same problem also exists in GA and PSO. Moreover, the FOA-GRNN model should be further tested and developed through its application in other fields of science, math, technology, and engineering.

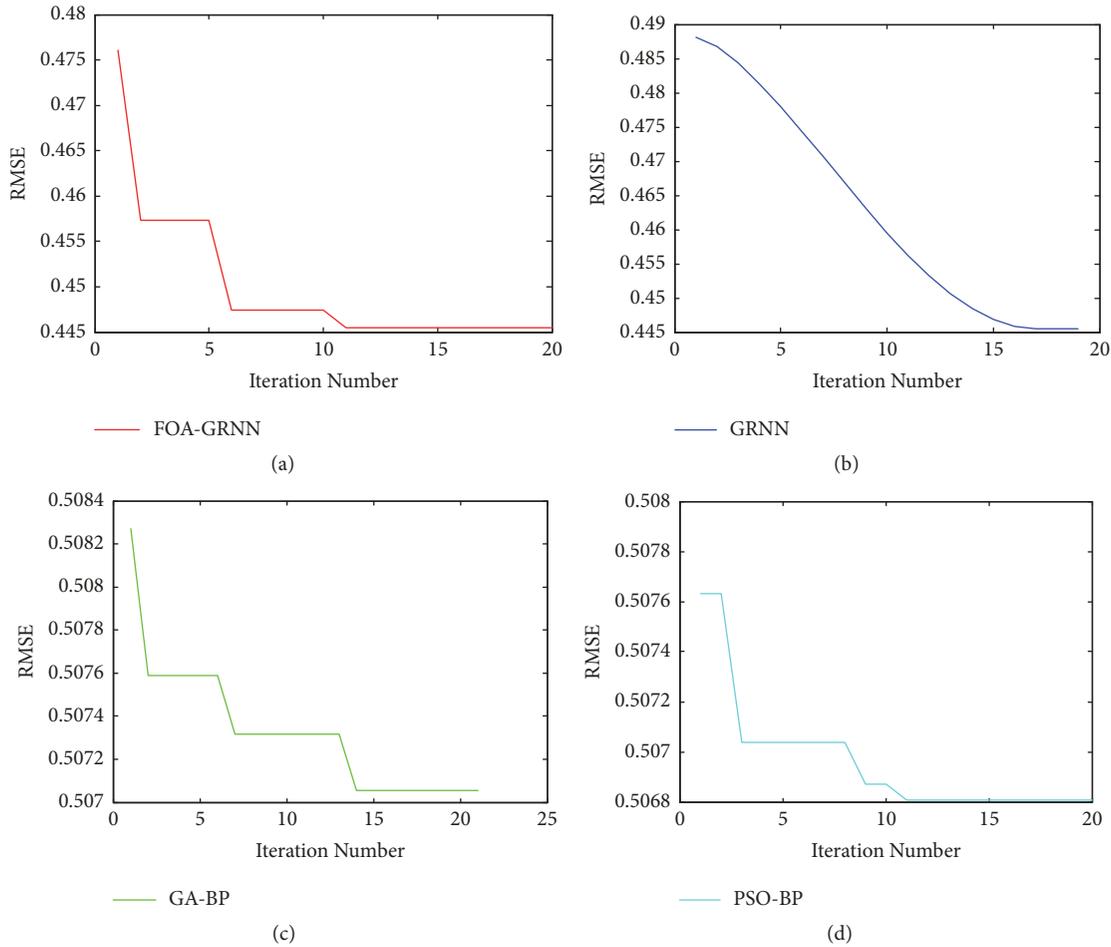


FIGURE 8: The optimization process of third testing function.

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 no conflicts of interest.

Acknowledgments

This research was funded by the National Social Science Fund of China (Grant no. 17BGL202).

References

[1] Z. Hua, Y. Wang, X. Xu, B. Zhang, and L. Liang, "Predicting corporate financial distress based on integration of support vector machine and logistic regression," *Expert Systems with Applications*, vol. 33, no. 2, pp. 434–440, 2002.

[2] J. M. Collins and M. R. Clark, "An application of the theory of neural computation to the prediction of workplace behaviour: an illustration and assessment of network analysis," *Personnel Psychology*, vol. 46, no. 3, pp. 503–524, 1993.

[3] S. H. Zanakis and I. Becerra-Fernandez, "Competitiveness of nations: A knowledge discovery examination," *European Journal of Operational Research*, vol. 166, no. 1, pp. 185–211, 2005.

[4] W. Luo and Z. Fu, "Application of generalized regression neural network to the agricultural machinery demand forecasting," *Applied Mechanics and Materials*, no. 278, pp. 2177–2181, 2013.

[5] R. Hu, S. Wen, Z. Zeng, and T. Huang, "A short-term power load forecasting model based on the generalized regression neural network with decreasing step fruit fly optimization algorithm," *Neurocomputing*, vol. 221, pp. 24–31, 2017.

[6] T. Chen and C. Yu, "Motion control with deadzone estimation and compensation using GRNN for TWUSM drive system," *Expert Systems with Applications*, vol. 36, no. 8, pp. 10931–10941, 2009.

[7] E. Orhan, F. Temurtas, and A. Ç. Tanrikulu, "Tuberculosis disease diagnosis using artificial neural networks," *Journal of Medical Systems*, vol. 34, no. 3, pp. 299–302, 2010.

[8] S. C. Chelgani and E. Jorjani, "Microwave irradiation pretreatment and peroxyacetic acid desulfurization of coal and application of GRNN simultaneous predictor," *Fuel*, vol. 90, no. 11, pp. 3156–3163, 2011.

[9] I. Ladhani, L. Houichi, L. Djemili, S. Heddami, and K. Belouz, "Modeling daily reference evapotranspiration (ET₀) in the north of Algeria using generalized regression neural networks (GRNN) and radial basis function neural networks (RBFNN):

- A comparative study," *Meteorology and Atmospheric Physics*, vol. 118, no. 3-4, pp. 163-178, 2012.
- [10] N. Ceryan, U. Okkan, and A. Kesimal, "Application of generalized regression neural networks in predicting the unconfined compressive strength of carbonate rocks," *Rock Mechanics and Rock Engineering*, vol. 45, no. 6, pp. 1055-1072, 2012.
- [11] F. Wang, B. Shen, S. Sun, and Z. Wang, "Improved GA and Pareto optimization-based facial expression recognition," *Assembly Automation*, vol. 36, no. 2, pp. 192-199, 2016.
- [12] G. Y. Li, F. Yan, and X. F. Guo, "Gray neural network algorithm improved by genetic algorithm," *Control Engineering of China*, vol. no. 20, pp. 934-937, 2013.
- [13] H. Zhao and S. Guo, "Annual energy consumption forecasting based on PSOCA-GRNN model," *Abstract and Applied Analysis*, vol. 2014, Article ID 217630, 11 pages, 2014.
- [14] J. J. Ma, J. X. You, and Z. Chen, "Grey neural network model based on modified particle swarm optimization algorithm and its application," *Journal of Tongji University (Natural Science)*, vol. 40, no. 5, pp. 740-743, 2012.
- [15] B. Dengiz, C. Alabas-Uslu, and O. Dengiz, "Optimization of manufacturing systems using a neural network metamodel with a new training approach," *Journal of the Operational Research Society*, vol. 60, no. 9, pp. 1191-1197, 2009.
- [16] R. S. Sexton, B. Alidaee, R. E. Dorsey, and J. D. Johnson, "Global optimization for artificial neural networks: A tabu search application," *European Journal of Operational Research*, vol. 106, no. 2-3, pp. 570-584, 1998.
- [17] M. Dorigo and L. M. Gambardella, "Ant colonies for the travelling salesman problem," *BioSystems*, vol. 43, no. 2, pp. 73-81, 1997.
- [18] M. Gazzah, B. Jaouachi, and F. Sakli, "Optimization of bagged denim fabric behaviors using the genetic algorithms and the ant colony optimization methods," *International Journal of Clothing Science and Technology*, vol. 27, no. 6, pp. 772-792, 2015.
- [19] W.-T. Pan, "A new fruit fly optimization algorithm: taking the financial distress model as an example," *Knowledge-Based Systems*, no. 26, pp. 69-74, 2012.
- [20] Y. Zhang, S. Na, J. Niu, and B. Jiang, "The influencing factors, regional difference and temporal variation of industrial technology innovation: Evidence with the FOA-GRNN model," *Sustainability*, vol. 10, no. 1, p. 187, 2018.
- [21] J. H. Holland, *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge, Mass, USA, 1992.
- [22] S. G. Kulkarni, A. K. Chaudhary, S. Nandi, S. S. Tambe, and B. D. Kulkarni, "Modeling and monitoring of batch processes using principal component analysis (PCA) assisted generalized regression neural networks (GRNN)," *Biochemical Engineering Journal*, vol. 18, no. 3, pp. 193-210, 2004.
- [23] H. B. Celikoglu, "Application of radial basis function and generalized regression neural networks in non-linear utility function specification for travel mode choice modelling," *Mathematical and Computer Modelling*, vol. 44, no. 7-8, pp. 640-658, 2006.
- [24] M. A. Fernández-Gámez, A. M. Gil-Corral, and F. Galán-Valdivieso, "Corporate reputation and market value: Evidence with generalized regression neural networks," *Expert Systems with Applications*, vol. 46, pp. 69-76, 2016.
- [25] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing*, vol. 1 and 2, MIT Press, Cambridge, Mass, USA, 1986.
- [26] Y. Huang, J. Zhang, X. Li, and L. Tian, "Thermal error modeling by integrating GA and BP algorithms for the high-speed spindle," *The International Journal of Advanced Manufacturing Technology*, vol. 71, no. 9-12, pp. 1669-1675, 2014.
- [27] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International of first Conference on Neural Networks*, pp. 1942-1948, December 1995.
- [28] L.-K. Shark and C. Yu, "Design of matched wavelets based on generalized Mexican-hat function," *Signal Processing*, vol. 86, no. 7, pp. 1451-1469, 2006.
- [29] F. Jin, D. Feng, and S. Gómez, "Image Registration Algorithm Using Mexican Hat Function-Based Operator and Grouped Feature Matching Strategy," *PLoS ONE*, vol. 9, no. 4, p. e95576, 2014.
- [30] L. A. Rastrigin, "Extremal control systems," *Theoretical Foundations of Engineering Cybernetics Series*, no. 3, 1974.
- [31] H. Mühlenbein, M. Schomisch, and J. Born, "The parallel genetic algorithm as function optimizer," *Parallel Computing*, vol. 17, no. 6-7, pp. 619-632, 1991.
- [32] A. O. Griewank, "Generalized descent for global optimization," *Journal of Optimization Theory and Applications*, vol. 34, no. 1, pp. 11-39, 1981.
- [33] M. Locatelli, "A note on the Griewank test function," *Journal of Global Optimization*, vol. 25, no. 2, pp. 169-174, 2003.



Hindawi

Submit your manuscripts at
www.hindawi.com

