Prediction of Network Security Situation Awareness based on an Improved Model Combined with Neural Network

Li Yuan (li54096@163.com)
Chongqing Electric Power College

Research

Keywords: Radial basis function neural network, network security situation awareness prediction, particle swarm optimization

Posted Date: April 26th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-407084/v1

Prediction of network security situation awareness based on an improved model combined with neural network

Li Yuan

Chongqing Electric Power College, Chongqing 400053, China

Corresponding address: No. 9, Dianli Fourth Village, Wulongmiao Community, Jiulongpo District, Chongqing 400053, China

Email: li54096@163.com

Abstract:

People always pay attention to the security of the network. This paper mainly analyzed the problem of network security situation prediction (NSSP). The Radial Basis Function (RBF) neural network was improved by the particle swarm optimization (PSO) algorithm, and a modified PSO (MPSO)-RBF algorithm was obtained, which was used as the prediction model. Then, the data from National Internet Emergency Center (CNCERT/CC) were used as the experimental data, and the MPSO-RBF algorithm was compared with RBF and PSO-RBF algorithms. The results showed that the MPSO-RBF algorithm could achieve convergence in about 50 times of iterations, showing a high calculation efficiency, and the mean absolute percentage error (MAPE) value, mean square error (MSE) value, and root-mean-square error (RMSE) value were small, 2.13%, 0.0005, and 0.0224, respectively, showing that the algorithm had good prediction performance. The results verify the reliability of the MPSO-RBF algorithm in NSSP, which is conducive to further improve network security.

Keywords: Radial basis function neural network, network security situation awareness prediction, particle swarm optimization

1. Introduction

With the development of Internet technology, the network has been applied in more and more fields, such as online shopping, electronic payment, smart home, and entertainment, which has brought great changes to people's production and lifestyle. However, at the same time, network security has

also been greatly challenged. Network attacks are becoming more complex and diverse. Although the current network security technologies, such as firewall [3], intrusion detection [4], and virtual private network (VPN), play a role in network security, they all have defects and are not suitable for the current more and more complex network environment; therefore, network security situation awareness (NSSA) appears [6, 7]. NSSA evaluates the current network state by processing the network security data, and on this basis, it can carry out network security situation prediction (NSSP) [8]. Zhang et al. [9] optimized the wavelet neural network (WNN) with the isolation niche genetic algorithm (INGA), carried out a simulation experiment, and found that the method had a higher prediction accuracy. Based on the recurrent neural network (RNN), Wei et al. [10] extracted features from the original time series data and verified on the RNN model after training. The results showed that the method had more accurate prediction results, although the training time was long. Aiming at the problem of long training time of the support vector machine (SVM) algorithm, Hu et al. [11] optimized the SVM algorithm with the cuckoo search (CS), carried out distributed training on MapReduce, and found that the method could effectively reduce the training time. Zhou et al. [12] solved the NSSP problem with the hidden belief rule base (HBRB), improved the prediction accuracy of the model based on the evidence reasoning rules, and verified the effectiveness of the new method by the case study. Based on neural network, this study analyzed the NSSP problem, designed an improved radial basis function (RBF) algorithm with the particle swarm optimization (PSO) algorithm, and verified the effectiveness of the algorithm in solving NSSP through experimental analysis, which makes a contribution to the better realization of network security.

2. Method

2.1 Network security situation awareness prediction

The idea of situation awareness first appeared in the military and was used for judging the military environment and situation, and then it was applied in fields such as transportation and medicine and extended to the field of network security. NSSA means to collect as many network security elements as possible through a series of technologies and establish corresponding evaluation and prediction models to help network managers deal with risks in time. NSSP means predicting the future network state and prevent network attacks based on historical situation assessment data.

The premise of prediction is that there are some rules between adjacent data points. The research shows that there is self-similarity in network traffic data. The prediction object of NSSP is network security situation (NSS) value, which is a series arranged in time order, and there are some rules between adjacent data points; therefore, NSSP is predictable, which is feasible.

At present, the methods used in NSSP are as follows: ①autoregressive moving average model [13]: based on stationary time series, it forecasts the future state, but has some requirements for the length of time series; ②grey theory [14]: based on grey correlation, it searches for the internal law of the system, but it has poor prediction performance for data with large fluctuation; ③time series [15]: it is based on the correlation between adjacent data, but many elements need to be considered in the process of modeling; ④neural network [16]: it takes security events as input and outputs the situation value to realize NSSP, but it is easy to fall into local convergence to affect the prediction effect.

2.2 Improved model of neural network

2.2.1 Radial basis function neural network

RBF neural network is a three-layer feedforward neural network [17]. The input layer transmits the input samples to the hidden layer, and the number of nodes is the dimension of the samples, which is written as $X = (x_1, x_2, \Lambda\ , x_i)$ ($i = 1,2,\Lambda\ , m$). The hidden layer trains the samples and adjusts the parameters at the same time. There are h nodes. The weight between the hidden layer and the input layer is w_{ij}, and the threshold is θ_j. The commonly used activation function is the Gaussian function, written as $\phi_j = \exp\left(-\frac{\|x - c_j\|}{2\delta_j^2}\right)$, where φ_j refers to the base function, c_j is the data center of the j-th node, and δ_j refers to the width parameter of the j-th node. The output layer responds to the input. The weight between the output layer and the hidden layer is v_{ij}, and the threshold is θ_k. The output of the j-th node is written as: $y_j = \sum_{i=1}^{h} v_{ij}\phi_j + \theta_k$. The specific training process of the RBF neural network-based NSSP model is as follows.

(1) The input data set $X = (x_1, x_2, \Lambda\ , x_N)$ is established. $A(L)$ is defined to accumulate the vector sum of NSS samples belonging to different classes. $B(L)$ is defined to register the number of NSS samples belonging to every class. L refers to the number of classes of NSS samples.

(2) Every sample is regarded as the possible data center, and the density indicator is calculated: $D_i = \sum_{j=1}^{N} \exp\left(-\frac{\left\| x_i - x_j \right\|^2}{(d_1/2)^2} \right)$, where d_1 is the radius of the neighbourhood that takes x_i as the center.

(3) For every sample point x_j, the distance between x_j and c_j is calculated. If $r \le d_2$ (d_2 is a threshold), then it is classified into the corresponding class. Moreover, let $A(1) = A(1) + x_j$ and $B(1) = B(1) + 1$.

(4) The unclassified sample is used as a new input sample X. The above steps are repeated until $B(L) < M$ (M is a set threshold). Finally, L classes of NSS are obtained.

(5) For each class, its center of gravity is calculated, $c_i = \frac{A(i)}{B(j)}$, until the neural network converges.

2.2.2 Particle swarm optimization algorithm

For the RBF neural network, data center c_j, width parameter δ_j, weight v_{ij}, and the number of nodes (h) in the hidden layer all have a significant impact on the performance of the neural network. Therefore, the above parameters are optimized by a modified particle swarm optimization (MPSO) algorithm in this study.

The PSO algorithm is an intelligent algorithm that simulates the foraging behavior of birds [18]. The possible solution of the algorithm is the position of particles, and individuals update

positions through adjusting the individual extremum p_{best} and the global extremum g_{best} to find out the optimal solution. It is assumed that there are M particles in a D-dimensional space. In the t-th time of iteration, the position and speed of the i-th particle can be written as:

$$x_{id}^{t} = \left\{x_{i1}^{t}, x_{i2}^{t}, \Lambda\ , x_{iD}^{t}\right\},$$

$$v_{id}^{t} = \left\{v_{i1}^{t}, v_{i2}^{t}, \Lambda\ , v_{iD}^{t}\right\}.$$

The update formulas can be written as:

$$v_{id}^{t+1} = wv_{id}^{t} + c_1 r_1\left(p_{best,i}^{t} - x_{id}^{t}\right) + c_2 r_2\left(g_{best}^{t} - x_{id}^{t}\right),$$

$$x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1},$$

where w is the inertia weight, c_1 and c_2 are acceleration factors, r_1 and r_1 are random numbers in (0, 1), $p_{best,i}^{t}$ is the individual optimal solution in the t-th time of iteration, and g_{best}^{t} is the globally optimal solution.

As the PSO algorithm is easy to fall into the local extremum and the convergence speed is slow, it is improved in aspects of inertia weight w and acceleration factors c_1 and c_2. Firstly, w is dynamically adjusted according to the following equation: $w(t) = w_{\max} - \frac{t}{t_{\max}} \times e^{\left(\frac{t_{\max}-t}{t_{\max}}\right)}\left(w_{\max} - w_{\min}\right)$, where $w_{\max} = 0.9$ and $w_{\min} = 0.3$. w can decrease with the increase of t, avoiding the PSO algorithm falling into the local optimum. Then, c_1 and c_2 are adjusted according to the following equations: $c_1(t) = c_{1\max} - \frac{t}{t_{\max}} \times e^{\left(\frac{t_{\max}-t}{t_{\max}}\right)}\left(c_{1\max} - c_{1\min}\right)$ and $c_2(t) = c_{2\max} - \frac{t_{\max} t}{t_{\max}} \times e^{\left(\frac{t}{t_{\max}}\right)}\left(c_{2\max} - c_{2\min}\right)$. c_1 can decrease with the increase of t, and c_2 can increase with the increase of t, improving the convergence speed of the PSO algorithm.

2.2.3 MPSO-RBF-based NSSP model

The parameters of the RBF neural network algorithm were optimized by the MPSO algorithm. The steps of obtaining the NSSP model based on the MPSO-RBF neural network algorithm are as follows.

(1)An NSS data sample set is established, preprocessed, and normalized. The data were adjusted to the numbers in $[0,1]$. The formula is $x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$, where $x_{\max}$ and $x_{\min}$ are maximum and minimum values.

(2) The data set is divided into training samples and test samples.

(3) The structure of the RBF neural network algorithm is determined.

(4) The parameters of the RBF neural network algorithm are encoded and mapped to particles in the particle swarm.

(5) The particle swarm is initialized, and the fitness value of particles is calculated.

(6) p_{best} and g_{best} of particles are updated.

(7) The position and speed of particles are updated.

(8) Whether the accuracy meets the requirements or whether it reaches the maximum number of iterations is determined. If it does, the algorithm ends, and the optimal parameters of the RBF neural network algorithm are output to establish a NSSP model.

3. Case analysis

3.1 Experimental data

Data in security weekly reports from National Internet Emergency Center (CNCERT/CC) from 2018 to 2020 were used as the experimental data. The basic situation of network security was evaluated with five indicators, and the situation was divided into five levels: excellent, good, medium, poor, and dangerous. To facilitate calculation, the five levels were represented by numbers 5-1. The data from 2018 to 2019 were used as training samples, numbered 1-104. The data in 2020 were taken as testing samples, numbered 105-156, as shown in Table 1.

Table 1 Experimental data

Number	Number of hosts infected with computer malicious program in China/10000	Total number of tampered websites in China/n	Total number of websites implanted with a back door in China/n	Number of counterfeit pages for Chinese websites/n	Number of new information security vulnerabilities/n	NSS value
1	20.1	2210	827	407	219	4
2	20.15	2117	773	419	257	4

3	23.9	2435	908	333	410	4
4	25.2	867	812	384	241	4
5	21.1	1992	758	318	382	4
……	……	……	……	……	……	……
100	56.83	1427	3631	690	402	3
101	49.2	461	2933	1216	481	4
102	57.3	1491	3119	1260	327	4
103	58.5	7138	2891	1992	230	4
104	60.5	6419	2775	1266	203	4
105	62.8	9829	2796	1620	215	4
106	58.9	8774	2777	1479	295	4
107	44.7	8218	2378	1236	354	4
108	53.3	9414	1747	169	221	4

109	54.6	7008	1681	84	221	4
110	50.5	1299	1395	1205	573	3
111	56	3363	1591	720	609	3
112	58.6	9636	1746	740	514	3
113	62.6	9140	1537	996	699	3
114	58.6	7529	1578	1570	554	3
……	……	……	……	……	……	……
152	50.3	4894	699	17727	360	4
153	57.1	4704	556	18114	266	4
154	60.9	4078	580	13861	293	4
155	75.8	4327	839	4374	214	4

156	52.3	4453	1498	4942	298	4

3.2 Evaluation index

(1) Mean absolute percentage error (MAPE): $$MAPE = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\left(\frac{y'_i - y_i}{y'_i}\right)} \times 100\%$$.

(2) Mean square error (MSE): $$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y'_i - y_i\right)^2$$.

(3) Root mean square error (RMSE): $$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y'_i - y_i\right)^2}$$.

n is the number of samples, and y'_i and y_i are the actual value and predicted value of the sample, respectively.

3.3 Prediction results

Firstly, the convergence speed of RBF, PSO-RBF, and MPSO-RBF neural network algorithms was analyzed. The results are shown in Figure 1.

It was seen from Figure 1 that the initial error of the RBF neural network algorithm was large, and there were some fluctuations in the change of the error. When the number of iterations reached about 350, the algorithm converged to the optimal error. After optimization by the PSO algorithm, the initial error of the RBF neural network algorithm reduced, and the convergence speed

significantly accelerated, reaching convergence after 200-300 times. The MPSO-RBF neural network algorithm designed in this study not only had a small initial error but also converged after 50 times and had a more stable error, which verified that the MPSO-RBF neural network algorithm had an advantage in convergence performance.

The prediction accuracy of different algorithms was compared. Taking samples numbered 105-115 as an example, the prediction results of different algorithms were compared, as shown in Figure 2.

It was seen from Figure 2 that the difference between the prediction result of the RBF neural network algorithm and the actual value was the biggest, and the change was not stable. After optimization by the PSO algorithm, the prediction result of the PSO-RBF neural network algorithm was improved, but there was still a gap with the actual value. The prediction value of the MPSO-RBF neural network algorithm nearly coincided with the actual value. The prediction error of different algorithms was calculated, and the results are shown in Table 2.

Table 2 Comparison of prediction error

Sample number	RBF neural network algorithm	PSO-RBF neural network algorithm	MPSO-RBF neural network algorithm
105	0.267	0.056	0.007
106	0.312	0.067	0.012
107	0.136	0.033	0.013

108	0.232	0.051	0.021
109	0.126	0.048	0.016
110	0.213	0.042	0.022
111	0.211	0.189	0.023
112	0.188	0.094	0.017
113	0.191	0.092	0.031
114	0.125	0.033	0.028
Maximum error	0.312	0.189	0.031
Minimum error	0.125	0.033	0.007
Average error	0.2001	0.0705	0.019

It was seen from Table 2 that the error of the RBF neural network algorithm was the largest, followed by the PSO-RBF neural network algorithm and the MPSO-RBF neural network function; the errors of the RBF neural network algorithm were all larger than 0.1, and the average error was 0.2001; the errors of the PSO-RBF neural network function were about 0.1, and the average error was 0.0705, which was 64.77% smaller than that of the RBF neural network algorithm; the errors of the MPSO-RBF neural network algorithm were smaller than 0.1, and the average error was only 0.019, which was 73.05% smaller than that of the PSO-RBF neural network algorithm. The above results demonstrated that the prediction result of the MPSO-RBF neural network algorithm was closer to the actual value, and the MPSO-RBF neural network algorithm had a good prediction performance.

The prediction performance of different algorithms was compared, and the results are shown in Table 3.

Table 3 Comparison of prediction performance

	MAPE	MSE	RMSE
RBF neural network algorithm	5.12%	0.0032	0.0566
PSO-RBF neural network algorithm	4.36%	0.0021	0.0458
MPSO-RBF neural network algorithm	2.13%	0.0005	0.0224

It was seen from Table 3 that the MAPE value of the RBF neural network algorithm was the largest, reaching 5.12%, the MAPE value of the PSO-RBF neural network algorithm was 0.76% smaller than that of the RBF neural network algorithm, and the MAPE value of the MPSO-RBF neural network algorithm was 2.23% smaller than that of the PSO-RBF neural network algorithm; the MSE value of the RBF neural network algorithm was the largest, followed by the PSO-RBF neural network algorithm and the MPSO-RBF neural network algorithm, and the comparison of the RMSE value was the same. It was found from Table 3 that the parameter optimization by the PSO algorithm effectively improved the prediction performance of the RBF neural network algorithm

and the MPSO algorithm improved the prediction performance of the RBF neural network, which showed significant advantages in the solving NSSP.

4. Discussion

The neural network realizes intelligent calculation to solve complex problems through imitating the structure and characteristics of neural networks of creatures with the help of mathematical and physical methods. It takes neurons as the basic unit, has feedforward and feedback structures, and has strong self-learning ability and good fault tolerance and stability, which has been widely used in image processing [19], fault diagnosis [20], financial analysis [21], and automatic control [22]. The classical neural network models include back propagation (BP) neural network [23], Hopfield neural network [24], RBF neural network, etc.

RBF neural network has a simple structure and good nonlinear approximation ability, which has been widely used in solving practical problems. This study improved the RBF neural network algorithm with the PSO algorithm and obtained the MPSO-RBF neural network algorithm to solve NSSP problem. From the perspective of the convergence performance, the MPSO-RBF neural network algorithm achieved convergence after about 50 times of iterations, and its convergence speed greatly improved compared with RBF and PSO-RBF neural network algorithms, which effectively improved the efficiency. Then, from the perspective of the prediction performance, it was found from Tables 2 and 3 that the MAPE, MSE, and RMSE values of the MPSO-RBF neural network algorithm were small, and the prediction performance of the MPSO-RBF neural network

algorithm was significantly better than that of the other two algorithms. It was concluded that the MPSO-RBF neural network algorithm was more suitable for solving the NSSP problem.

Although some achievements have been made in the research of NSSP in this study, there are still some shortcomings that need to be solved in future works:

(1) comparing the performance of more neural networks in solving NSSP problem;

(2) further optimize the PSO algorithm to improve the performance of the RBF neural network algorithm better;

(3) apply the MPSO-RBF neural network algorithm in the actual network environment.

5. Conclusion

Based on the RBF neural network, this study analyzed its usability in the NSSP problem, designed an MPSO-RBF model, and carried out experiments with the weekly safety report published by CNCERT/CC as the data set. The results showed that the MPSO-RBF neural network algorithm designed in this study had faster convergence and smaller prediction error compared with RBF and PSO-RBF neural network algorithms, with an MAPE value of 2.13%, a MSE value of 0.0005, and an RMSE value of 0.0224, which were smaller than the other two algorithms. The MPSO-RBF neural network algorithm shows good performance in solving the NSSP problem and can be further promoted and applied in practice.

Declarations

Availability of data and material

The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

Competing interests

None.

Funding

Not applicable.

Authors' contributions

LY designed research, performed research, analyzed data, and wrote the paper.

Acknowledgements

Not applicable.

References

[1] P. N. Dawadi, D.J. Cook, M. Schmitter-Edgecombe, Automated Clinical Assessment from Smart home-based Behavior Data. IEEE J. Biomed. Health 20, 1188 (2016). doi: 10.1109/JBHI.2015.2445754

[2] H. Lin, Z. Yan, Y. Chen, L. Zhang, A Survey on Network Security-Related Data Collection Technologies. IEEE Access, 2018, 1-1 (2018). doi: 10.1109/ACCESS.2018.2817921

[3] Y. Ariyanto, B. Harijanto, V. A. H. Firdaus, S. Arief, Performance analysis of Proxmox VE firewall for network security in cloud computing server implementation. IOP Conf. Ser. Mater. Sci. Eng. 732, 012081 (2020). doi: 10.1088/1757-899X/732/1/012081

[4] R. Venkatesh, R. Muthalagu, Network security prediction model using neural networks. J. Phys. Conf. Ser. 1706, 012167 (7pp) (2020). doi: 10.1088/1742-6596/1706/1/012167

[5] E. Conrad, S. Misenar, J. Feldman, Domain 4: Communication and Network Security (Designing and Protecting Network Security). CISSP Study Guide (Third Edition), 2016, 219-291 (2016).

[6] D. M. Zhao, J. X. Liu, Study on Network Security Situation Awareness based on Particle Swarm Optimization Algorithm. Comput. Ind. Eng. 2018, S036083521830007X (2018). doi: 10.1016/j.cie.2018.01.006

[7] G. Xu, Y. Cao, Y. Ren, X. Li, Z. Feng, Network Security Situation Awareness Based on Semantic Ontology and User-defined Rules for Internet of Things. IEEE Access, 2017, 1-1 (2017). doi: 10.1109/ACCESS.2017.2734681

[8] B. Zhu, Y. Chen, Y. Cai, Three Kinds of Network Security Situation Awareness Model Based

on Big Data. Int. J. Netw. Secur. 21, 115-121 (2019). doi: 18163548-201901-201901180002-201901180002-115-121

[9] H. Zhang, Q. Huang, F. Li, J. Zhu, A network security situation prediction model based on wavelet neural network with optimized parameters. Digit. Commun. Netw. 2016, 139-144 (2016). doi: 10.1016/j.dcan.2016.06.003

[10] W. Feng, Y. Wu, Y. Fan, A new method for the prediction of network security situations based on recurrent neural network with gated recurrent unit. Int. J. Intell. Comput. Cybern. 11, IJICC-06-2017-0066- (2018). doi: 10.1108/IJICC-06-2017-0066

[11] J. Hu, D. Ma, C. Liu, Z. Shi, H. Yan, C. Hu, Network security situation prediction based on MR-SVM. IEEE Access, PP, 1-1 (2019). doi: 10.1109/ACCESS.2019.2939490

[12] Z. J. Zhou, G. Y. Hu, B. C. Zhang, C. H. Hu, Z. G. Zhou, P. L. Qiao, A Model for Hidden Behavior Prediction of Complex Systems Based on Belief Rule Base and Power Set. IEEE T. Syst. Man Cy. S. 2017, 1-7 (2017). doi: 10.1109/TSMC.2017.2665880

[13] N. Garg, K. Soni, T. K. Saxena, S. Maji, Applications of AutoRegressive Integrated Moving Average (ARIMA) approach in time-series prediction of traffic noise pollution. Noise Control Eng. J. 63, 182-194 (2015). doi: info:doi/10.3397/1/376317

[14] L. Shen, Z. Wen, Network security situation prediction in the cloud environment based on grey neural network. J. Comput. Method. Sci. Eng. 19, 153-167 (2019). doi: 10.3233/JCM-180873

[15] Y. Shi, R. Li, Y. Zhang, X. N. Peng, An immunity-based time series prediction approach and its application for network security situation. Intel. Serv. Robot. 8, 1-22 (2015). doi:

10.1007/s11370-014-0160-z

[16] J. P. Cai, L. Qu, G. Chen, J. Yang, Application of an Improved Recurrent Neural Network in Network Security Situation Prediction. Adv. Mater. Res. 951, 173-176 (2014). doi: 10.4028/www.scientific.net/AMR.951.173

[17] W. Jia, D. Zhao, L. Ding, An optimized RBF neural network algorithm based on partial least squares and genetic algorithm for classification of small sample. Appl. Soft Comput. 2016, 373-384 (2016). doi: 10.1016/j.asoc.2016.07.037

[18] J. B. Scriver, Washington International Business Report. Tien Tzu Hsueh Pao/acta Electr. Sin. 43, 1538-1544 (2015). doi: 10.3969/j.issn.0372-2112.2015.08.011

[19] X. Hu, G. Feng, S. Duan, L. Liu, A Memristive Multilayer Cellular Neural Network With Applications to Image Processing. IEEE T. Neur. Net. Lear. 2016, 1-13 (2016). doi: 10.1109/TNNLS.2016.2552640

[20] R. Ahmed, M. E. Sayed, S. A. Gadsden, J. Tjong, S. Habibi, Automotive Internal-Combustion-Engine Fault Detection and Classification Using Artificial Neural Network Techniques. IEEE T. Veh. Technol. 64, 21-33 (2015). doi: 10.1109/TVT.2014.2317736

[21] R. Hafezi, J. Shahrabi, E. Hadavandi, A bat-neural network multi-agent system (BNNMAS) for stock price prediction: Case study of DAX stock price. Appl. Soft Comput. 29, 196-210 (2015). doi: 10.1016/j.asoc.2014.12.028

[22] Y. V. Bodyanskiy, A. K. Tyshchenko, A. A. Deineko, An evolving radial basis neural network with adaptive learning of its parameters and architecture. Autom. Control Comp. S. 49, 255-260

(2015). doi: 10.3103/s0146411615050028

[23] C. Huang, Y. Zhao, W. Yan, Q.Q. Liu, J.M. Zhou, A New Method for Predicting Crosstalk of Random Cable Bundle Based on BAS-BP Neural Network Algorithm. IEEE Access, 8, 1-1 (2020). doi: 10.1109/ACCESS.2020.2969221

[24] H. Wang, Y. G. Yu, G. G. Wen, S. Zhang, J. Z. Yu, Global stability analysis of fractional-order Hopfield neural networks with time delay. Neurocomputing, 154, 15-23 (2015). doi: 10.1016/j.neucom.2014.12.031

Figure 1 Comparison of convergence speed

Figure 2 Comparison of prediction results

Figures

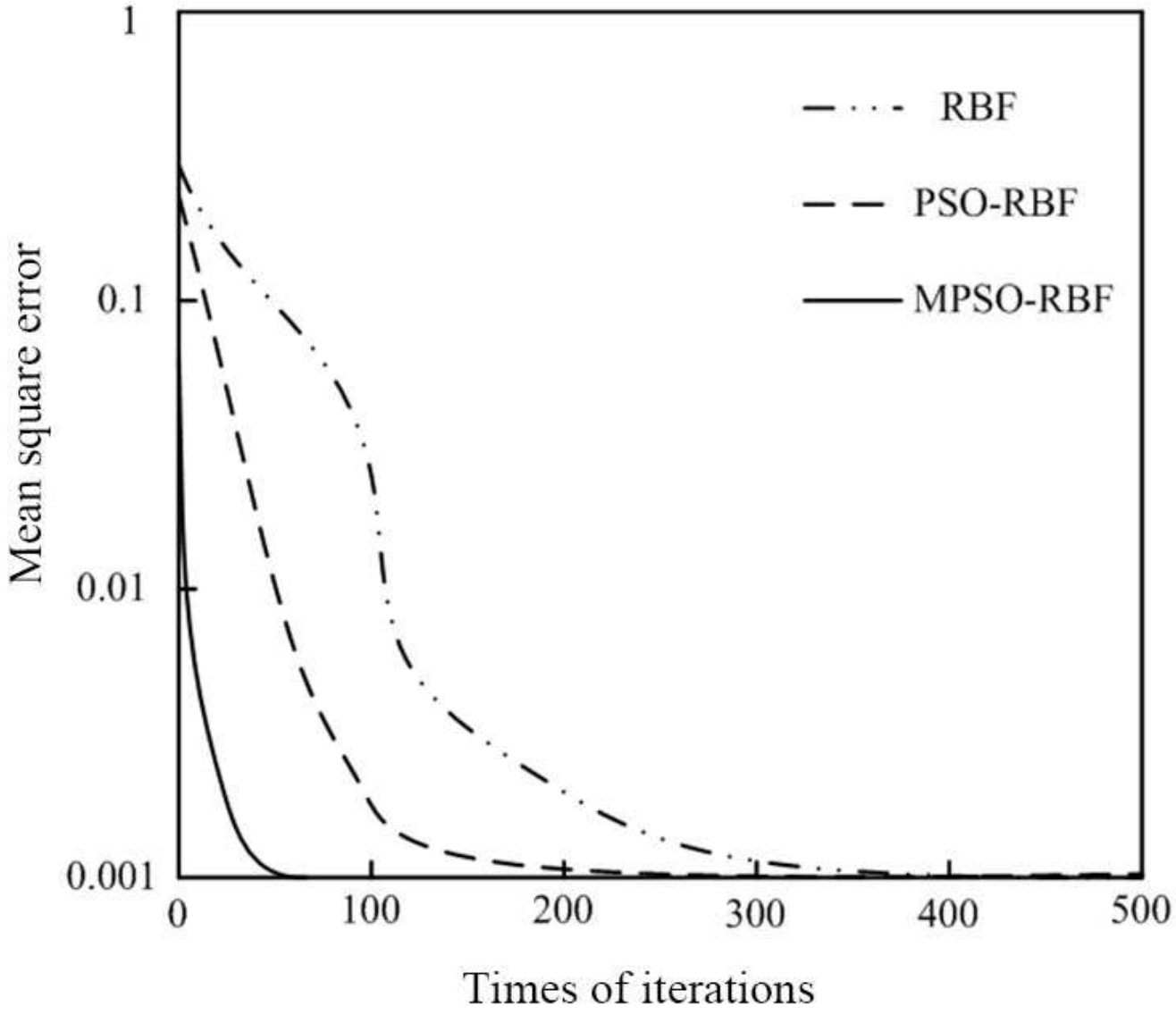

Figure 1

Comparison of convergence speed

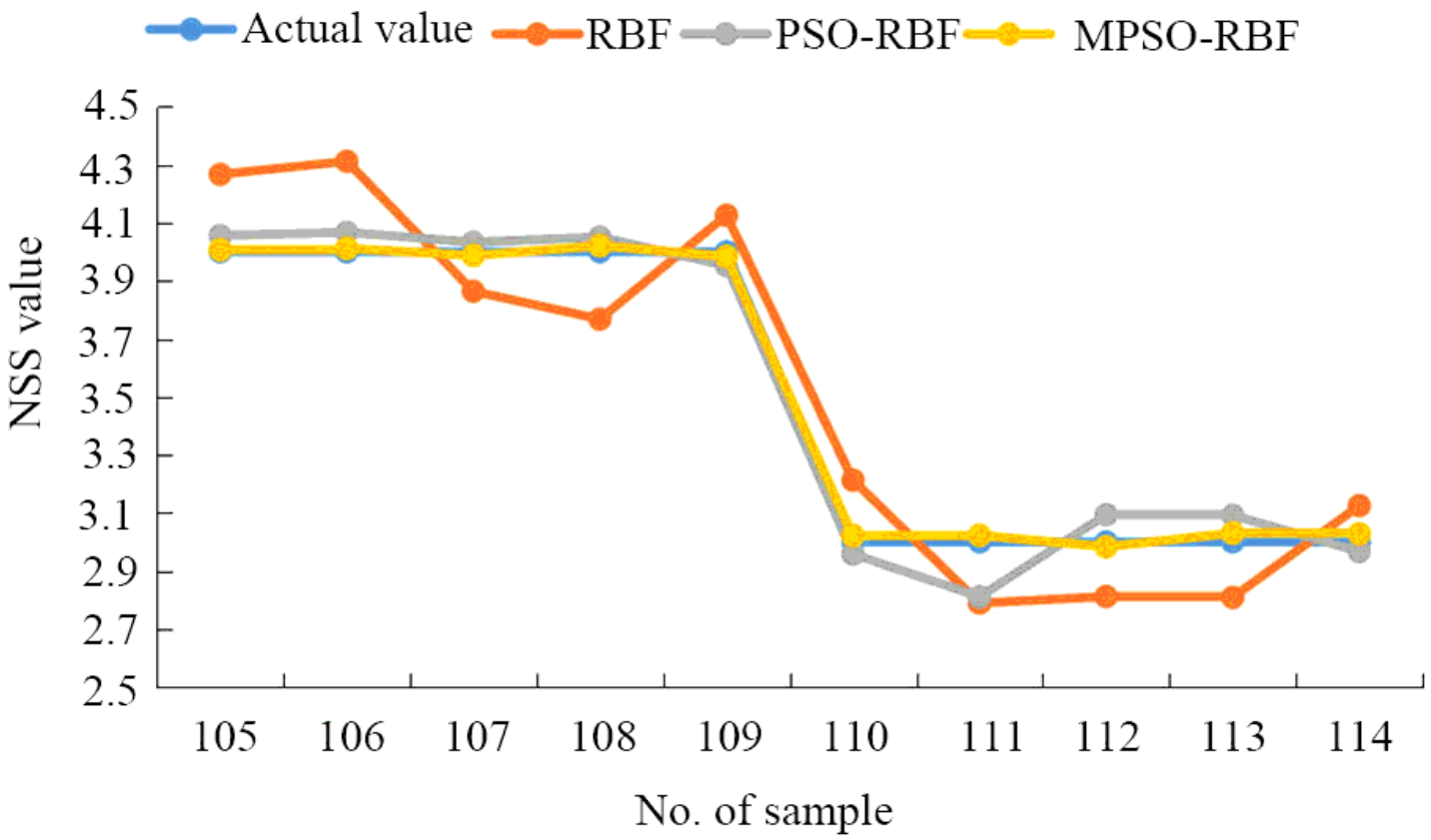

Figure 2

Comparison of prediction results