Received: 31 January 2019 | Revised: 21 May 2019 | Accepted: 16 September 2019

DOI: 10.4218/etrij.2019-0040

ORIGINAL ARTICLE

ETRI Journal WILEY

A network traffic prediction model of smart substation based on IGSA-WNN

Xin Xia[1,2] | **Xiaofeng Liu**[2] | **Jichao Lou**[3]

[1]College of Computer Information Technology, Wuhan Institute of Shipbuilding Technology, Wuhan, China

[2]Department of Information Technology, Wenzhou Vocational and Technical College, Wenzhou, China

[3]School of Computer Science, Wuhan University, Wuhan, China

Correspondence
Xin Xia, College of Computer Information Technology, Wuhan Institute of Shipbuilding Technology, Wuhan, China.
Email: xinxinfavor@gmail.com

Funding information
This work was supported by the Scientific Research Project of Zhejiang Provincial Department of Education, China (project number: Y201738745).

The network traffic prediction of a smart substation is key in strengthening its system security protection. To improve the performance of its traffic prediction, in this paper, we propose an improved gravitational search algorithm (IGSA), then introduce the IGSA into a wavelet neural network (WNN), iteratively optimize the initial connection weighting, scalability factor, and shift factor, and establish a smart substation network traffic prediction model based on the IGSA-WNN. A comparative analysis of the experimental results shows that the performance of the IGSA-WNN-based prediction model further improves the convergence velocity and prediction accuracy, and that the proposed model solves the deficiency issues of the original WNN, such as slow convergence velocity and ease of falling into a locally optimal solution; thus, it is a better smart substation network traffic prediction model.

KEYWORDS

analysis, gravitational search algorithm, modeling, network traffic prediction, smart substation, traffic measurement, visualization, wavelet neural network

1 | INTRODUCTION

Since the beginning of the 21st century, a wave of smart grids has been initiated in the global power field, and methods of building new smart grids have become a common research goal of power researchers worldwide [1]. Both the transmission and interaction of security-protection- and measurement-control-related information depends on network communication, so whether the network of smart substations can operate stably and reliably will directly affect the security of the entire smart substation. Meanwhile, the prediction of network traffic can effectively help power workers to warn of emergencies and to take countermeasures; thus, guaranteeing the security of smart substations [2,3]. The main issue aimed to be solved by the work reported in this paper is how to effectively predict the network traffic of a smart substation to provide relevant reference information for state analysis, emergency monitoring, and illegal intrusion detection of smart substation networks.

Presently, the model utilized in the network traffic prediction of smart substations mainly refers to the traffic prediction model of computer networks. In Ref. [4], a prediction model was established based on the possibility that networks are utilized at a certain time and place by users, and the model can collect and analyze the behavior characteristics of a considerable amount of network users; thus, achieving the prediction of network traffic. However, the design of this kind of model is highly difficult due to the slow computation velocity and low prediction accuracy; other researchers [5] have proposed application of the space-time compressed sensing method to mobile network traffic prediction, which is based on the space- and time-constraint matrices, and also refers to the variation characteristics of the period stability of mobile networks.

In recent years, the theories and technologies of data mining and machine learning (ML) have been increasingly improved and applied to multiple fields, and have achieved good results. Therefore, data mining [6–8] and ML [9–11] algorithms have been gradually introduced into the field of network traffic prediction. Support vector machines (SVMs) have also been applied to the field of network traffic prediction and have achieved better prediction results [12,13]. Based on the traditional SVM, researchers have further proposed the use of a variety of optimized SVM models for network traffic prediction. Additionally, we have previously researched the back-propagation (BP) neural network [14,15], currently a very popular ML algorithm, and successfully introduced it into the prediction of network traffic. Generally, in relevant experiments in which the BP neural network is utilized as a prediction method, the input information is the traffic value at the period of the first k sections, while the output information is the predicted traffic value at one or several periods after the kth section. Presently, by continuous iterative optimization of the parameter values of the neural network, the optimal network architecture is consequently obtained for traffic prediction. Historical research shows that there is a certain correlation between the values before and after the network traffic. Network traffic was predicted by the Elman neural network in [16], and comparison results also prove that the prediction effect of the model is better than that of the BP neural network.

In accordance with the long-term sampling analysis of network traffic, we found that the network traffic data of a smart substation have the characteristics of periodicity, randomization, and burstiness. Related research shows that the relation between network traffic and affecting factors is very complex, and the prediction accuracy of traditional methods is not high. The wavelet neural network (WNN) has been proved as a good traffic prediction model and can achieve an accurate description of network traffic [17–19]. With both the time-frequency local characteristics of a wavelet transform (WT) and the fitting ability of an artificial neural network (ANN) [20–22], a WNN makes full use of the advantages of both items and can achieve the fitting of arbitrary functions by arbitrary precision, but it still has deficiencies, such as slow training velocity and ease of falling into a locally optimal solution.

In this paper, we propose some algorithms to improve the performance of the network traffic prediction model. The contributions of this paper are as follows.

1. As the use of an exponential function as a gravitational constant equation in the standard gravitational search algorithm (GSA) causes faster decline of the algorithm's global optimization capability, we propose a novel method of optimizing the gravitational constant equation of the GSA.
2. As a novel method of optimizing the velocity equation of the GSA, we introduce chaotic operators to solve the issues of premature convergence and a local minimum. To combine particle swarm optimization (PSO), we present an improved velocity equation.
3. We apply the improved GSA to the initialization of the WNN [23–25]. Therefore, we propose the IGSA-WNN algorithm to enhance the convergence velocity and prediction accuracy.

The remainder of this paper is organized as follows: The methods are given in Section 2, after which a solution approach is presented in Section 3. The experiment is presented in Section 4. Finally, the conclusions are presented in Section 5.

2 | IMPROVEMENT OF GSA

2.1 | Optimization of gravitational constant equation

In the iterative optimization process of the GSA, the gravitational constant $G(t)$ can directly affect the magnitude of the external force acting on particles, further affecting the accelerated velocity and velocity change of particles, and finally affecting the location update of particles. That is to say, the selection of a gravitational constant equation that conforms more with GSA characteristics can improve the optimization performance of the GSA; thus, allows the network traffic prediction model of a smart substation to achieve better prediction results. In accordance with the characteristics of the GSA, at the earlier stage of its iterative optimization, the GSA must explore the solution space; that is, search for the optimal solutions in a wider range. In this case, the search velocity should be faster and the corresponding gravitational constant should be larger; while at the later stage of iterative optimization, the GSA must develop the solution space, that is, approximate the optimal solutions in a narrower range. In this case, the search velocity should be slower and the corresponding gravitational constant should be smaller. The gravitational constant equation utilized by the standard GSA is

$$G(t) = G_0 e^{-a\frac{t}{T}} \tag{1}$$

where $G(t)$ represents the gravitational constant, G_0 is the initial gravitational constant value, a is utilized to adjust the changing velocity of the gravitational constant, and t and T represent the current and total number of iterations, respectively. In the gravitational constant equation, the corresponding $G(t)$ decreases very fast, which makes the global optimization capability of the algorithm decline faster and impacts the prediction performance of the traffic prediction model based on this algorithm.

To enable the GSA to achieve better optimization results and, thus, improve the prediction ability of the traffic prediction model, in this paper, we optimize the gravitational constant equation in the standard GSA and propose the following gravitational constant equation:

$$G(t) = G_0\left(1-\sqrt{\frac{t}{T}}\right). \tag{2}$$

The optimized gravitational constant equation solves the issue that the use of the exponential function as a gravitational constant equation in the standard GSA causes the faster decline of the algorithm's global optimization capability, guarantees its global optimization effect, and provides great help to the optimization of the network traffic prediction model based on a WNN for the subsequent utilization of the algorithm.

2.2 | Optimization of velocity equation

The standard GSA still has the issues of premature convergence and a local minimum, and the location update process of particles only considers the information of the current particle itself, which subsequently leads to poor prediction precision of the network traffic prediction model of a smart substation that uses the standard GSA for optimization. To improve the traffic prediction model's prediction performance, in this paper, we propose the following methods to optimize the velocity equation in view of the abovementioned issues of the GSA.

(a) Introduction of chaotic operators [26,27]. In the standard GSA, a random number is utilized in the particle velocity equation to solve the issues of premature convergence and a local minimum in the algorithm. The chaotic model shows nonlinear dynamic behaviors and, meanwhile, has ergodicity and randomization. Moreover, previous research indicates that the search performance by using chaotic variables is better than that of random search. Therefore, chaotic operators can be introduced into the particle velocity equation to optimize the GSA; the added chaotic operators are

$$c_{i+1} = vc_i(1-c_i) \quad i=1,2,\ldots n. \tag{3}$$

In (3), c_i is a random number in the value range [0, 1], and the value of v is taken as 4.

(b) Combine the PSO [28–30]. The PSO algorithm is a commonly used optimization algorithm that obtains the optimal solution by location iteration of particles in solution space. However, unlike the GSA, which only considers the current location of particles in the particle location update, the PSO algorithm also considers the historical optimal location of the particle itself, as well as the optimal location of the entire particle swarm. The velocity equation of the PSO algorithm is expressed as

$$v_i(t+1)=wv_i(t)+b_1\text{rand}_j(p_{\text{best}}-x_i(t))+b_2\text{rand}_k(g_{\text{best}}-x_i(t)), \tag{4}$$

$$b_1=(b_{1f}-b_{1i})\times\frac{t}{T}+b_{1i}, \tag{5}$$

$$b_2=(b_{2f}-b_{2i})\times\frac{t}{T}+b_{2i}, \tag{6}$$

where w represents inertia mass; rand_j and rand_k are random numbers in the value range [0, 1]; the value ranges of b_1 and b_2 are both [0-1]; $x_i(t)$ represents the location of the ith particle at moment t; p_{best} is the optimum location of the current particle i among all historical locations before moment t; and g_{best} is the optimum location of the entire particle swarm among all historical locations before moment t. The PSO algorithm controls the impact of the particle information itself and swarm information on the velocity by b_1 and b_2 in (4). Specifically, b_{1i} and b_{2i} are the starting values of b_1 and b_2, whereas b_{1f} and b_{2f} are the stop values of b_1 and b_2. The optimization research on the PSO algorithm shows that the utilization of (5) and (6) to adjust b_1 and b_2 can well balance the exploration ability and developmental ability of the algorithm; that is, at the earlier stage of iterative optimization, when enabling the information of the particle itself to better impact the particle velocity update, the particle can search at a larger space. At the later stage of iterative optimization, when enabling the particle swarm information to better impact the particle velocity update, the particle can move toward the global optimal region.

In accordance with the PSO algorithm, in this paper, we improve the optimization performance of the GSA by introducing the historical optimal location of the particle itself, as well as the optimal location of the entire particle swarm into the velocity equation of the GSA, balancing the exploration ability and development ability of the algorithm in accordance with (5) and (6), and propose the following velocity equation by introducing the abovementioned chaotic operators:

$$\begin{aligned}v_i(t+1)=&[\text{rand}_i\times v_i(t)+\xi(c_i-0.5)]+a_i(t)+b_1\text{rand}_j(p_{\text{best}}(t)-x_i(t))\\&+b_2\text{rand}_k(g_{\text{best}}(t)-x_i(t)).\end{aligned} \tag{7}$$

In (7), ξ is used to control the chaotic scope, and $a_i(t)$ represents the accelerated velocity of the particle, the specific values of which are shown in the computation process of the original GSA. The other parameter values are shown in the first description.

3 | SOLUTION APPROACH

3.1 | Data-normalization process

The traffic of a smart substation has the characteristics of periodicity, randomization, and burstiness, among others. If the network traffic data obtained from crude sampling is directly analyzed and predicted, it will disturb the training process of the network traffic prediction model, and impact the training velocity and prediction accuracy. Therefore, we must first normalize network traffic data:

$$x' = x'_{\min} - \left[\frac{x - x_{\min}}{x_{\max} - x_{\min}}\left(x'_{\max} - x'_{\min}\right)\right] \tag{8}$$

where $x'_{\max}$ is set as 0.9 and $x'_{\min}$ as 0.8; x represents the original network traffic data obtained from direct sampling; x' is the network traffic data obtained after normalization; $x_{\min}$ is the minimum value of network traffic in the original network traffic data obtained from direct sampling; $x_{\max}$ is the maximum value of network traffic in the original network traffic data obtained from direct sampling.

3.2 | Data reconstruction and classification

The form of network traffic data in a certain period of time is expressed as

$$x = \left\{x_1, x_2, \ldots, x_t\right\} \tag{9}$$

where x represents the network traffic data sequence after normalization treatment at a certain continuous time and x_t is the network traffic data at moment t.

For network traffic prediction, in this paper we utilize the following function as the prediction model function:

$$\hat{y} = f(x_1, x_2, \ldots, x_t) \tag{10}$$

where $\hat{y}$ represents the predicted value of network traffic.

It is advisable to take network traffic data at k continuous periods as input values. The prediction of network traffic data at the next period is achieved in accordance with these input values, then, the reconstruction method for the sample data is as shown in Table 1.

3.3 | Initialization of WNN-related parameters

3.3.1 | Input layer

The number of neuron nodes of the input layer is determined by the issues to be solved and the form of input data.

TABLE 1 Reconstruction method of network traffic data

	Network traffic input	Network traffic output
1	$x_1, x_2, \ldots, x_k$	y_1
2	$x_2, x_3, \ldots, x_{k+1}$	y_2
3	$x_3, x_4, \ldots, x_{k+2}$	y_3
4	$x_4, x_5, \ldots, x_{k+3}$	y_4
…	…	…
n	$x_n, x_{n+1}, \ldots, x_{n+k-1}$	y_n

In the network traffic prediction model of a smart substation, the number of nodes of the input layer depends on the number of periods in the traffic sequence selected to predict the next period after reconstruction of the data obtained from crude sampling. For example, in this paper, we utilize the network traffic data of the first k time intervals to predict the network traffic data of the latter m time intervals; the number of nodes of the input layer should then be set as k.

3.3.2 | Hidden layer

According to previous research experience, the following equation can be used to help select the number of relatively better neurons of the hidden layer n_1 [31]:

$$n_1 = \sqrt{n+m} + a \tag{11}$$

where n and m represent the numbers of nodes of the input layer and output layer, respectively, and a is a constant with a value between 1 and 10.

3.3.3 | Output layer

The number of neuron nodes of the output layer is determined by the issues to be solved and the actual demands of experimenters. Generally, if the traffic of the latter m time intervals is predicted in accordance with the traffic sequence of the first k time intervals, the number of neuron nodes of the output layer should be selected as m.

3.3.4 | Activation function

The hidden layer and output layer of a WNN both utilize "M-P neurons," so corresponding activation functions should be selected, and the wavelet basis function should be selected for the hidden layer.

3.3.5 | Design of the initial value of parameter

Generally, the initial values of weight, scalability, and shift factors, for example, of the parameters are taken randomly.

3.4 | Network Traffic Prediction Based on IGSA-WNN Model

Step 1: Initialization of IGSA parameter values

First, the particle swarm is encoded, and each particle represents a set of weight, scalability factor, and shift factor. Then, the initialization parameters of the IGSA are determined, including the magnitude of universal gravitational constant, particle scale, as well as iterations of the algorithm, among others.

Step 2: Determination of fitness function

The fitness function is an error function related to the target output value and actual output value; the fitness function [32] selected in this model is given by

$$\text{fit}(t) = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(y_{ji} - y'_{ji}\right)^2 \tag{12}$$

where y_{ji} is the theoretical output value of the ith sample at the jth output layer, y'_{ji} is the actual output value of the ith sample at the jth output layer, n is the sample data volume for training, and m is the number of neuron nodes of the output layer.

Step 3: Iterative optimization by IGSA

The location of a particle is continuously adjusted by the change of the fitness-function value until the given terminal condition is reached and the particle at the optimal location is decoded, then the initial parameter value of the neural network can be obtained.

Step 4: Training of traffic prediction model based on WNN

The optimal initial weight, scalability, and shift factors obtained by the IGSA iterative optimization are used as the initial parameter values in the neural network, and training samples are used to conduct corresponding training for the WNN until the given iterations or given error requirements are reached. Then, the training is stopped.

Step 5: Utilization of trained prediction model for network traffic prediction.

The trained WNN prediction model with all the connection weights, scalability factors, and shift factors determined is utilized to conduct prediction experiments.

A flowchart of the IGSA-WNN model is depicted in Figure1.

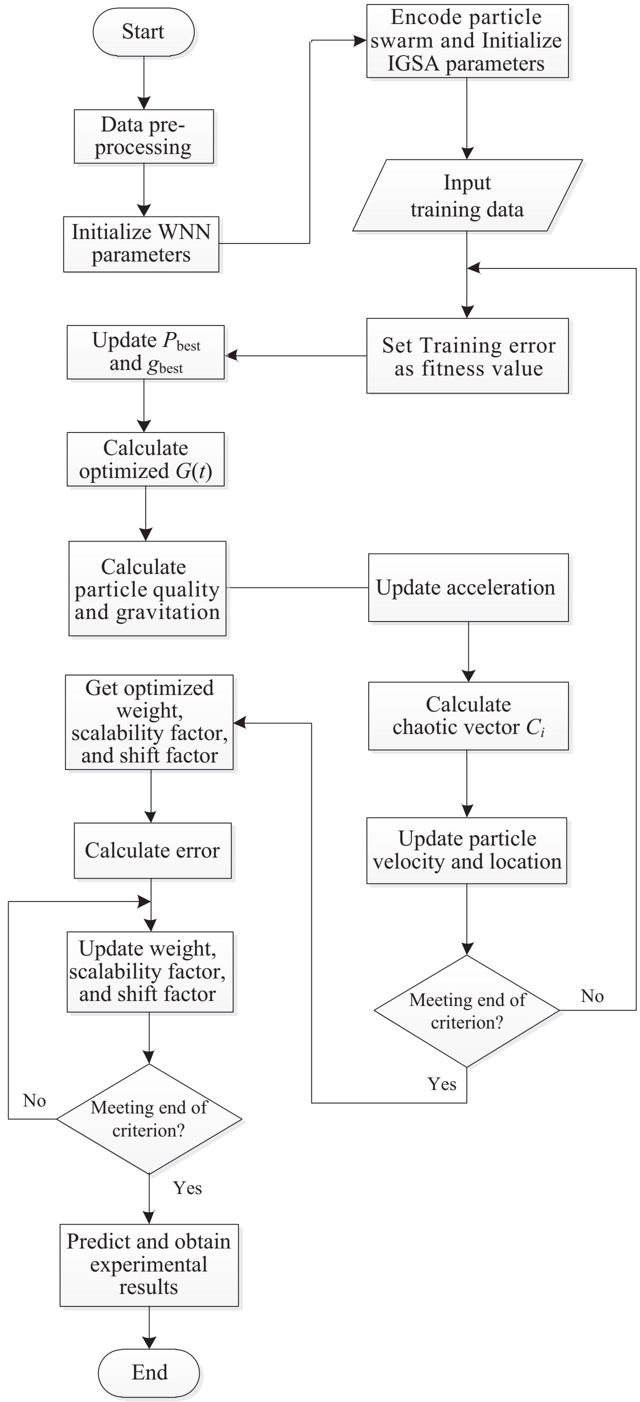


FIGURE 1 Flowchart of the IGSA-WNN model

4 | EXPERIMENT

4.1 | Experiment design

4.1.1 | Experimental data acquisition

The division of network traffic of a smart substation by the division method of a virtual local-area network (VLAN) is the key to utilization of the same network by the GOOSE and SV messages in the smart substation for transmission [33]. The smart substation from which the data were acquired in this experiment separates the high-tension transmission line from the three sides of the main transformer and utilizes an independent switch to set different network segments. The GOOSE network utilizes a two-layer network model for communication of the switching value and other signals; that is, this kind of network provides an independent switch for each independent network segment and a corresponding public switch for all voltage classes. The independent switch equipped for each independent network segment and the public switch equipped for the corresponding voltage classes are interconnected by an uplink port.

The traffic data for the network traffic prediction experiments of a smart substation at this time were acquired by the Simple Network Management Protocol on a network switch of a smart substation in Hubei Province. The time interval for the sampling of network traffic is 1 second, and the continuous acquisition lasts for 20 hours to obtain 72 000 pieces of traffic data. To enhance the experimental reliability, we divided the acquired data into 10 groups, with each group containing 7200 pieces of traffic data. The 10 groups of data are utilized to conduct prediction experiments for the three kinds of traffic prediction models.

4.1.2 | Data preprocessing

Normalization of traffic data

First, the original network traffic data were normalized in accordance with (8). Owing to the large data volume of the experiments, we took the first 300 seconds of the first group and normalized the data sequence of network traffic, as shown in Figure 2.

Reconstruction of traffic data

In this experiment, the traffic data of five continuous time intervals were utilized to predict the traffic data of the next time interval. Then, 5000 pieces of sample data were obtained after reconstruction treatment of the first group of normalized traffic data, as shown in Table 2.

Classification of sample data

We chose fivefold cross-validation. The 5000 pieces of sample data were classified, and, specifically, 4000 pieces of the sample data were utilized as training data to conduct training for the network traffic prediction model while the other 1000 pieces of sample data were used as predicted data to analyze the model's prediction accuracy.

4.1.3 | Evaluation standard for prediction performance

There are numerous evaluation standards for the network traffic prediction model; we utilized the mean-square error (MSE) and equalization coefficient (EC) to evaluate the prediction performance of the prediction model. Both of these error functions can well-reflect the fitting degree of the predicted and actual values.

The MSE equation is expressed as

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{x}_i - x_i)^2 \tag{13}$$

and the EC equation as

$$\mathrm{EC} = 1 - \frac{\sqrt{\sum_{i=1}^{n}(\hat{x}_i - x_i)^2}}{\sqrt{\sum_{i=1}^{n}\hat{x}_i^2} + \sqrt{\sum_{i=1}^{n}x_i^2}} \tag{14}$$

where $\hat{x}_i$ represents the predicted value of traffic at a certain time interval, while x_i is the corresponding actual traffic value of $\hat{x}_i$. Generally, it can be considered that the smaller the value of the MSE, the higher the fitting degree of the predicted and

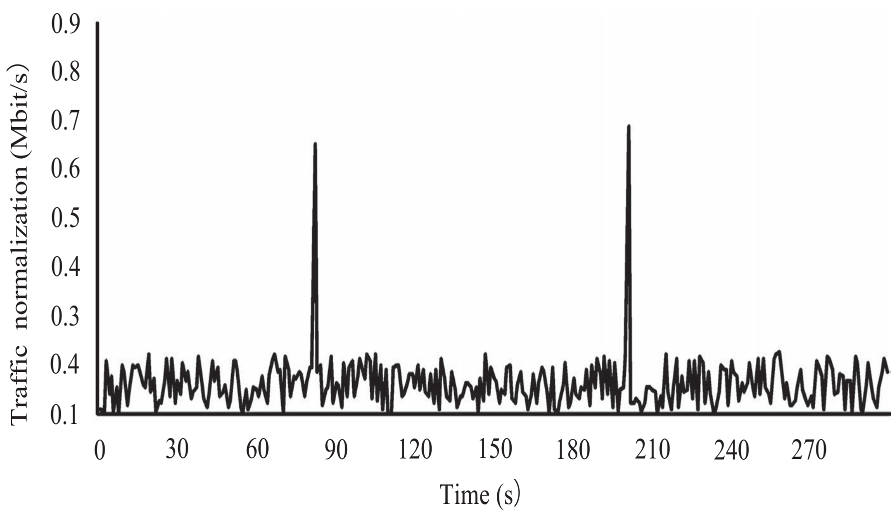


FIGURE 2 Data sequence of normalized network traffic

TABLE 2 Sample data of network traffic after reconstruction

No.	Input1	Input2	Input3	Input4	Input5	Output
1	0.68937	0.20623	0.16224	0.11328	0.16960	0.17931
2	0.20623	0.16224	0.11328	0.16960	0.17931	0.14327
3	0.16224	0.11328	0.16960	0.17931	0.14327	0.19237
4	0.11328	0.16960	0.17931	0.14327	0.19237	0.11217
5	0.16960	0.17931	0.14327	0.19237	0.11217	0.17031
6	0.17931	0.14327	0.19237	0.11217	0.17031	0.21826
7	0.17892	0.11319	0.21239	0.16821	0.17128	0.23112
8	0.11319	0.21239	0.16821	0.17128	0.23112	0.15128
9	0.21239	0.16821	0.17128	0.23112	0.15128	0.20863
10	0.16821	0.17128	0.23112	0.15128	0.20863	0.16272

actual values; the value of the EC is within [0,1], and it generally can be considered that the higher the value of the EC, the higher the fitting degree of the predicted and actual values. It is considered that the prediction results are better when EC > 0.9.

4.1.4 | Setting of prediction model parameters

Setting of WNN parameters

Topology of neural network. In this experiment, we selected the neural network model of a single-hidden layer to reduce the complexity of the neural network model and reduce the computation time. The network traffic data of the first five continuous time intervals were selected in the experiment to predict the network traffic data of the next time interval after the first five continuous time intervals, so the number of nodes of the input layer was set as 5 and that of the output layer as 1. For the number of nodes of the hidden layer, we first utilized the empirical equation suggested in (11) to estimate the value range of the number of nodes of the hidden layer, then, the number of neuron nodes of the hidden layer finally obtained was found to be 7 as a result of repeated trials.

Activation function. For the selection of the activation function of the hidden layer, we utilized the Morlet wavelet basis function as the activation function of the hidden layer of the neural network because the Morlet wavelet has shown a very strong anti-interference nature while having very high time-frequency resolution in past network traffic prediction experiments. The Morlet wavelet basis function [34] can be expressed as

$$h(t) = \cos(1.75^t) e^{\frac{-t^2}{2}}. \tag{15}$$

For the activation function of the output layer, the wavelet basis function has continuous and smooth characteristics, and the function value obtained after input signal processing is distributed in the range [0, 1]. Obviously, the distribution of output values when this function is utilized for the activation function of the neural network is comparatively reasonable, so the activation function of neurons of the output layer is a Sigmoid function [35], expressed as

$$f(x) = \frac{1}{1 + e^{-x}}. \tag{16}$$

Other parameters. In this experiment, the target value of the network error was set as 0.00001, the learning rate η as 0.08, and the number of neural network iterations as 2000.

Setting of GSA and IGSA parameters

For the GSA and IGSA parameters, we set a in the gravitational constant equation of the GSA as 20 and ε in the velocity equation of the IGSA as 3; in addition, $b_{1i} = 1$, $b_{2i} = 0$, $b_{1f} = 0$, and $b_{2f} = 1$. Moreover, the gravitational constant G_0 was set as 100, the quantity of particle swarms as 30, the maximum iterations of the algorithm as 1000, and the fitness function as the MSE function.

4.2 | Experimental results and analysis

After the initial GSA, IGSA, and WNN parameters were determined, for each group of traffic data the particle swarm was first initialized, then iterative optimization was conducted through use of the GSA and IGSA, respectively. Finally, the optimal initial parameter value was obtained. At this point, the optimal initial parameter values (weight, shift factor, and scalability factor) obtained from the GSA and IGSA optimization were utilized to construct the WNN prediction model, the training sample data utilized to conduct repeated training for WNN,and then the trained WNN prediction model was utilized for the predicted sample data. Meanwhile, the WNN prediction model was utilized to directly predict the network traffic. In this way, the prediction results of three kinds of network traffic prediction models were obtained.

To better show the prediction results of the model, we randomly selected 10 pieces of continuous predicted data in the first group of experiments, and the prediction results of the traffic prediction models based on the WNN, GSA-WNN, as well as IGSA-WNN, obtained separately by calculation, are shown in Table 3.

We separately utilized 10 groups of traffic sample data to conduct experiments for the three kinds of prediction models and calculate their MSEs for the obtained prediction results; the obtained MSEs of the three kinds of models are shown in Table 4 and Figure 3.

The ECs were calculated for the obtained prediction results, and the obtained ECs for the three kinds of models are shown in Table 5 and Figure 4.

As can be seen from Table 4 and Figure 3, the MSE of the prediction model based on IGSA-WNN is smaller than those of the other two kinds of prediction models. As can be seen from Table 5 and Figure 4, the EC of the prediction model based on IGSA-WNN is greater than those of the other two kinds of prediction models. Then, as can be seen from the MSE and EC values of the three kinds of models, the prediction accuracy of the prediction model based on IGSA-WNN is superior to both those of the prediction model based on WNN and that based on GSA-WNN.

At the same time, we separately recorded the number of iterations of 10 groups of experimental data for the three kinds of prediction models, as shown in Table 6.

TABLE 3 Prediction results based on three kinds of traffic prediction models

No.	Actual value	WNN	GSA-WNN	IGSA-WNN
1	0.18318	0.19365	0.17358	0.17567
2	0.10037	0.09003	0.09057	0.10807
3	0.11469	0.10404	0.12454	0.10703
4	0.21322	0.22442	0.22276	0.22083
5	0.69821	0.70942	0.70791	0.70581
6	0.13471	0.14518	0.14423	0.14223
7	0.10231	0.09173	0.11212	0.10988
8	0.20094	0.21206	0.21060	0.19327
9	0.16429	0.15337	0.15443	0.17201
10	0.21057	0.22215	0.22016	0.21814

TABLE 4 MSE comparison of three kinds of prediction models (units: 10^{-5})

Sample group	WNN	GSA-WNN	IGSA-WNN
1	11.7961	9.3969	5.7962
2	12.9808	10.4097	6.8442
3	11.8083	9.0738	6.2546
4	19.6377	11.5416	6.5227
5	10.9898	10.1970	6.0987
6	14.0933	10.0695	6.2451
7	16.9473	10.6492	7.2870
8	12.4752	9.6134	6.5664
9	19.8692	12.6423	7.3045
10	10.9149	9.1731	6.1464

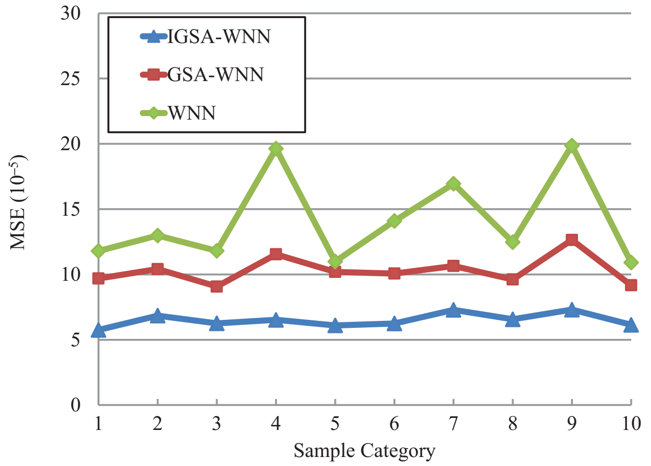


FIGURE 3 MSE comparison of three kinds of prediction models

TABLE 5 EC comparison of three kinds of prediction models

Sample group	WNN	GSA-WNN	IGSA-WNN
1	0.9667	0.9706	0.9773
2	0.9687	0.9711	0.9765
3	0.9668	0.9716	0.9764
4	0.9663	0.9709	0.9781
5	0.9695	0.9710	0.9775
6	0.9625	0.9702	0.9771
7	0.9669	0.9715	0.9765
8	0.9675	0.9714	0.9764
9	0.9673	0.9703	0.9773
10	0.9681	0.9717	0.9768

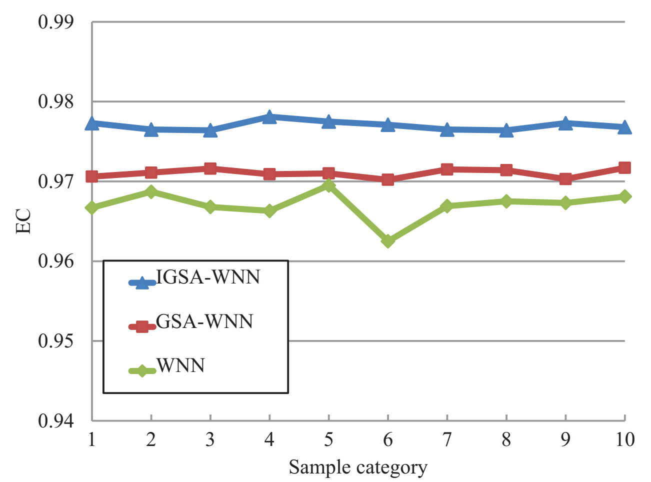


FIGURE 4 EC comparison of three kinds of prediction models

TABLE 6 Iteration times of three kinds of prediction models

Sample group	WNN	GSA-WNN	IGSA-WNN
1	1932	624	467
2	1929	619	461
3	1925	620	458
4	1918	621	463
5	1932	623	464
6	1919	618	462
7	1934	620	465
8	1931	619	464
9	1935	617	467
10	1928	619	468
Average	1928	620	464

The experimental results show that the average number of iterations of the 10 groups of experimental data based on WNN, GSA-WNN, and IGSA-WNN are 1928, 620, and 464, respectively. Thus, the convergence velocity of the IGSA-WNN-based prediction model is faster than that of the other two models.

Previous research indicates that the prediction model based on WNN has deficiency issues, such as slow convergence velocity and ease of falling into a locally optimal solution. The experimental results show that the traffic prediction model based on IGSA-WNN, as proposed in this paper, can effectively accelerate the convergence velocity of the network and improve the accuracy of traffic prediction. In summary, the prediction performance of the prediction model based on IGSA-WNN is superior to those based on WNN and GSA-WNN.

5 | CONCLUSIONS

In view of the deficiencies of the prediction model based on WNN, such as slow network training velocity and ease of falling into a locally optimal solution, in this paper, we proposed the use of a swarm intelligence algorithm with global optimization capability to optimize a traditional WNN. Meanwhile, in view of the deficiencies of the GSA, we proposed combining the optimization of the gravitational coefficient equation with PSO algorithm optimization, then, combined the IGSA with a WNN, and proposed the resulting network traffic prediction model based on IGSA-WNN. Finally, we conducted comparison experiments for the prediction model based on IGSA-WNN, a traditional prediction model based on WNN, as well as a prediction model based on GSA-WNN. The experimental results show that the traffic prediction model based on IGSA-WNN can effectively accelerate the convergence velocity of the network and improve the accuracy of traffic prediction.

ORCID

Xin Xia https://orcid.org/0000-0001-8892-6366

REFERENCES

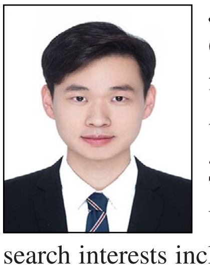
1. N. Rajagopal and K. V. Prasad, *Process framework for Smart Grid implementation*, in Proc. Int. IEEE Innovative Smart Grid Technologies-Asia (Bangalore, India), 2013, pp. 1–5.
2. T. Sidhu, M. Kanabar, and P. Parikh, *Configuration and performance testing of IEC 61850 GOOSE*, in Proc. IEEE Int. Conf. Adv. Pow. Sys. Auto. Protection (Beijing, China), 2011, pp. 1384–1389.
3. J. Ko et al., *A novel network modeling and evaluation approach for security vulnerability quantification in substation automation systems*, IEICE Trans. Info. Syst. **E96-D** (2013), no. 9, 2021–2025.
4. Z. Kun et al., *Group-based search in unstructured peer-to-peer, networks*, in Proc. Int. IEEE Global Telecomm. Conf. (Honolulu, HI, USA), 2009, pp. 1–6.
5. M. Roughan et al., *Spatio-temporal compressive sensing and internet traffic matrices (extended version)*, IEEE ACM Trans. Netw. **20** (2012), no. 3, 662–676.
6. T. Eterovic et al., *Data mining meets network analysis: traffic prediction models*, in Proc. Int. Convention. Inf. Commun. Technol., Elec. Microelec. (Opatija, Croatia), 2014, pp. 1479–1484.
7. W. H. Lee et al., *Discovering traffic bottlenecks in an urban network by spatiotemporal data mining on location-based services*, IEEE Trans. Intell. Trans. Syst. **12** (2011), no. 4, 1047–1056.
8. J. Wang et al., *Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach*, in Proc. Int. IEEE Conf. Comput. Commun. (Atlanta, GA, USA), 2017, pp. 1–9.
9. Y. Zang et al., *Wavelet transform processing for cellular traffic prediction in machine learning networks*, in Proc. IEEE China Summit and Int. Conf. Signal Inf. Process (Chengdu, China), 2015, pp. 458–462.
10. G. Fusco et al., *Short-term traffic predictions on large urban traffic networks: Applications of network-based machine learning models and dynamic traffic assignment models*, in Proc. Int. Conf. Models Technol. Intell. Trans. Syst. (Budapest, Hungary), 2015, pp. 93–101.
11. S. Suthaharan, *Big data classification: Problems and challenges in network intrusion prediction with machine learning*, Performance Evaluation Rev. **41** (2014), no. 4, 70–73.
12. Q. M. Wang, A. W. Fan, and H. S. Shi, *Network traffic prediction based on improved support vector machine*, Int. J. Syst. Assurance Eng. Manag. **8** (2017), 1976–1980.
13. M. Deshpande and P. R. Bajaj. *Performance analysis of support vector machine for traffic flow prediction*, in Proc. Int. Conf. Global Trends Signal, Process. Inf. Comput. Commun (Jalgaon, India), 2016, pp. 126–129.
14. J. C. Li et al., *A link prediction method for heterogeneous networks based on BP neural network*, Physica A **495** (2018), 1–17.
15. H. S. Wang, Y. N. Wang, and Y. C. Wang, *Cost estimation of plastic injection molding parts through integration of PSO and BP neural network*, Expert Syst. Appl. **40** (2013), no. 2, 418–428.
16. G. Ren et al., *A modified Elman neural network with a new learning rate scheme*, Neurocomput. **286** (2018), 11–18.
17. J. Adamowski and H. F. Chan, *A wavelet neural network conjunction model for groundwater level forecasting*, J. Hydrol. **407** (2011), no. 1–4, 28–40.
18. H. J. Yang and X. Hu, *Wavelet neural network with improved genetic algorithm for traffic flow time series prediction*, Optik **127** (2016), no. 19, 8103–8110.

19. M. Santhosh, C. Venkaiah, and D. M. V. Kumar, *Ensemble empirical mode decomposition based adaptive wavelet neural network method for wind speed prediction*, Energy Convers Manage **168** (2018), 482–493.
20. M. Sameti, M. A. Jokar, and F. R. Astaraei, *Prediction of solar stirling power generation in smart grid by GA-ANN model*, Int. J. Comput. Applicat. Technol. **55** (2017), no. 2, 147–157.
21. W. Wang et al., *Forecasting daily streamflow using hybrid ANN models*, J. Hydrol. **324** (2006), no. 1–4, 383–399.
22. P. M. Watson and K. C. Gupta, *Design and optimization of CPW circuits using EM-ANN models for CPW components*, IEEE Trans. Microw. Theory Techn. **45** (1997), no. 12, 2515–2523.
23. X. Yuan et al., *Short-term wind power prediction based on LSSVM-GSA model*, Energy Convers Manag. **101** (2015), 393–401.
24. Z. Jadidi et al., *Flow-based anomaly detection using neural network optimized with GSA algorithm*, in Proc. Int. Conf. Distribut. Comput. Syst. (Philadelphia, PA, USA), 2013, pp. 76–81.
25. P. Haghbayan, H. Nezamabadi-Pour, and S. Kamyab, *A niche GSA method with nearest neighbor scheme for multimodal optimization*, Swarm and Evolutionary Comput. **35** (2017), 78–93.
26. G. Xu, C. Xiu, and Z. Wan, *Hysteretic chaotic operator network and its application in wind speed series prediction*, Neurocomput. **165** (2015), 384–388.
27. Y. He et al., *Reservoir flood control operation based on chaotic particle swarm optimization algorithm*, Appl. Math. Model **38** (2014), no. 17–18, 4480–4492.
28. B. G. S. Dhas and S. N. Deepa, *A hybrid PSO and GSA-based maximum power point tracking algorithm for, PV systems*, in Proc. IEEE Int. Conf. Computat. Intell. Comput. Research (Enathi, India), 2013, pp. 1–4.
29. S. Jayaprakasam, S. K. A. Rahim, and C. Y. Leow, *PSOGSA-Explore: A new hybrid metaheuristic approach for beampattern optimization in collaborative beamforming*, Appl. Soft Comput. J. **30** (2015), 229–237.
30. R. K. Khadanga and J. K. Satapathy, *Unified power flow controller based damping controller design: - A hybrid PSO-GSA approach*, in Proc. Int. Conf. Energy, Power Environ. (Shillong, India), 2015, pp. 1–6.
31. D. Yang et al., *Application of BP neural network in image compression under the matlab*, in Proc. Int. Conf. Comput. Syst., Electron. Contr. (Dalian, China), 2018, pp. 1081–1086.
32. B. A. Solhmirzaei, S. Azadi, and R. Kazemi, *Road profile estimation using wavelet neural network and 7-DOF vehicle dynamic systems*, J. Mech. Sci. Technol. **26** (2012), no. 10, 3029–3036.
33. M. G. Kanabar, T. S. Sidhu, and M. R. D. Zadeh, *Laboratory investigation of IEC 61850-9-2-based busbar and distance relaying with corrective measure for sampled value loss/delay*, IEEE Trans. Power Delivery **26** (2011), no. 4, 2587–2595.
34. H. Sun, Y. Xiang, and Y. Guo, *Classification Method of EEG Signals Based on Wavelet Neural Network*, in Int. Conf. Bioinform. Biomedical Eng. (Beijing, China), 2009, pp. 1–4.
35. L. Yu, B. Chen, and J. Xiao, *An integrated system of intrusion detection based on rough set and wavelet neural network*, in Proc. Int. Conf. Natural Comput. (Haikou, China), 2007, pp. 194–199.

AUTHOR BIOGRAPHIES

Xin Xia received his BS and MS degrees in Computer Science and Technology from Wuhan University, Wuhan, China, in 2003 and 2006, respectively. From 2010 to 2013, he worked for ZTE Corp. as a senior pre-research engineer. From 2013 to 2019, he has been with the Department of Information Technology, Wenzhou Vocational & Technical College, Wenzhou, China. He is currently working with the College of Computer Information Technology, Wuhan Institute of Shipbuilding Technology, Wuhan, China. His research interests include computer network and communication, machine learning, and computer vision.

Xiaofeng Liu received his BS degree in Mathematics and his MS degree in Computer Science and Technology from Xiamen University in Xiamen, China, in 2002 and 2005, respectively. He is now the director of Software Technology Specialty at the Department of Information Technology, Wenzhou Vocational & Technical College, Wenzhou, China. His current research interests include computer software theory, machine learning, and deep learning.



Jichao Lou received his BS degree in Computer Science and Technology from Anhui University of Technology, Anhui, China, in 2016, and his MS degree of Computer Science and Technology from Wuhan University, Wuhan, China, in 2018. His current research interests include computer network and communication, machine learning, and deep learning.