Network traffic prediction of the optimized BP neural network based on Glowworm Swarm Algorithm

Haitao Li

College of Information Technology, Shangqiu Normal University, Shangqiu, People’s Republic of China

ABSTRACT

In order to improve the neural network structure and parameters set methods, on the basis of Glowworm Swarm Algorithm and the BP neural network, a Glowworm Swarm Algorithm to optimize the BP neural network algorithm is proposed. The algorithm uses the Glowworm Swarm Algorithm to obtain the better initial weights and thresholds of the network, to make up for the random defects of the BP neural network in the selection of connection weights and threshold and display the mapping ability of the generalization of BP neural network, and also make the BP neural network has fast convergence and strong learning ability. Applying the algorithm to the measured network flow algorithm and compared with the BP neural network and the Glowworm Swarm Algorithm to optimize the BP neural network, the simulation results show that the algorithm has higher forecast accuracy, thus proves the feasibility and effectiveness of the algorithm in the field of the forecast.

1. Introduction

With the rapid development of Internet and network technology, the network scale is increasing, and the network topology structure is also more and more complex, the problem of network performance and network service quality has become very prominent. The user wants to throughput the network to get better quality, more security services; at the same time, the network service providers hope to improve the network’s controllability and management, so as to improve the utilization ratio of the network. Therefore, in the case of limited network resources, the establishment of network traffic forecast model, real-time forecasting of the network, timely to make control or adjustment, will greatly improve the quality of network performance and network services.

Network traffic has the characteristics of nonlinear, long-relevant, time-varying, self-similarity and unexpected (Beran, Sherman, & Taqqu, 1995; Park & Willinger, 2000; Roughan, Veitch, & Abry, 2000), is a typical nonlinear time series. In recent years, the research on network traffic prediction has a lot of research methods. The traditional network traffic prediction Model are Interrupted Poisson Process (IPP) Model (Heffes & Lucantoni, 1986), Markov Modulated Fluid Model (MMFM) Model (Maglaris, Anastassiou, Sen, Karlsson, & Robbins, 1988), Auto Regressive Moving Average Model (ARMA) (Grunenfelder, Cosmas, Manthorpe, & Odinma-Okafor, 1991), the network traffic prediction methods are based on the linear model. However, with the rapid development of network scale, the network traffic in real environment is very complex, and linear method to predict the nonlinear network traffic cannot accurately characterize the flow of these characteristics, so the prediction results are not satisfactory, and the accuracy is not high. The existing research has proved that the neural network system with nonlinear structure can approximate any nonlinear function (Hornik, Stinchcombe, & White, 1989). Because of its powerful learning and multi mapping ability, the neural network can construct a nonlinear model with the input and output data. It can accurately describe the nonlinear relationship among various factors, so it is widely used in network traffic prediction (Atiya, Alym, & Parlosa, 2005; Hou & Wu, 2015; Hussein, 2001; Yu, 2013; Zhao, 2013). But in practical applications, the neural network has some inevitable defects, such as slow convergence speed, the potential of local extreme points and the poor ability to adapt to the network, especially the initial weights and thresholds of the neural network have a great influence on the performance of (Li & Li, 2011). Glowworm Swarm optimization algorithm is simulated firefly mating or foraging behaviour and puts forward a new swarm intelligence algorithm (Krishhand & Ghose, 2009). It not
only has a strong global optimization ability, and do not require the gradient of the objective function, with ease of use, robustness and easy realization characteristics, very suitable for optimizing the parameters of the neural network.

From the angle of the nonlinear time series, this paper puts forward a kind of the Glowworm Swarm Algorithm to optimize the BP neural network algorithm (IGSOBPNN). The algorithm uses Glowworm Swarm Algorithm to get better initial weights and thresholds, the method uses Glowworm Swarm Algorithm to compensate BP neural network connection weights and thresholds random selection of defects, which can play a mapping of BP neural network generalization ability, and can make the BP neural network has faster convergence and strong learning ability. Using this method to measured network traffic modelling and forecasting analysis, the simulation results show that the algorithm has higher accuracy of prediction.

2. Glowworm Swarm Algorithm

The basic Glowworm Swarm Algorithm is a new swarm intelligence optimization algorithm proposed by Krishnanad in 2005 (Krishhand & Ghose, 2009), it is the search and optimization process simulation into firefly individuals attract and moving process, by solving the problem of the objective function to quantify the individual merits of the firefly’s location. In the Glowworm Swarm Algorithm, each firefly is distributed within the definition of the objective function space, the individual has its own decision-making radius and its own portable fluorescent powder, firefly brightness is determined by the value of the target function of its location, the greater the intensity of the above point where the firefly has better target function value, which can attract more fireflies in the direction of the movement. Due to the radius of each firefly has its own decision-making, decision-making radius will be affected by the adjacent fireflies, at the same time when the surrounding the small number of fireflies, fireflies decision-making radius will increase, so that they can attract more fireflies around. When the fireflies around data more, decision-making radius is smaller. In the end, most of the fireflies gather in multiple locations, reach the extreme value point (that is the optimal objective function value).

Assuming that the number of the firefly is \(N\), and which the firefly \(i\) location is \((x_i, y_i)\), the corresponding target function is \(f(x_i, y_i)\) and the place of firefly fluorescein value is \(T_i\), the decision radius of each firefly is updated as shown in the formula (1).

\[
R_d^i(t + 1) = \min\{R_s, \max\{0, R_d^i(t) + \beta(n_t - |N_i(t)|)\}\} \quad (1)
\]

In the formula, \(R_d^i(t + 1)\) is the i firefly in the \(t + 1\) generation of the decision-making radius (that is the decision radius), \(R_s\) is perception radius, \(\beta\) is Control parameter, \(n_t\) is the threshold for the number of neighbours around the firefly, \(N_i(t)\) is the number of the firefly which has high luciferase in the decision radius, which the \(N_i(t)\) formula can be showed with the (2) formula:

\[
N_i(t) = \{j : |x_j(t) - x_i(t)|| < R_d^j, l_j(t) < l_i(t)\} \quad (2)
\]

In the formula, \(x_j(t)\) represents the position of the i firefly in generation \(t\), \(l_j(t)\) represents the value of the fluorescent value of the i firefly in generation \(t\). The distance between the adjacent firefly is within the range of the visual field radius of \(R_d^i\).

The rate of the firefly i to its neighbour firefly j is calculated by the formula (3):

\[
p_{ij}(t) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} l_k(t) - l_i(t)} \quad (3)
\]

Firefly location update calculated by the formula (4):

\[
x_i(t + 1) = x_i(t) + s \left( \frac{x_j(t) - x_i(t)}{||x_j(t) - x_i(t)||} \right) \quad (4)
\]

In the formula, \(s\) is the moving step.

After the firefly \(i\) moved to the new location, the fluorescein values will be recalculated according to the formula (5):

\[
l_i(t + 1) = (1 - \rho)l_i(t) + \gamma f(x_i(t + 1)) \quad (5)
\]

In the formula, \(l_i(t + 1)\) is the \(t + 1\) iteration, the value of firefly luciferase; \(\rho \in (0,1)\) is a constant, represents the neighbourhood threshold related to the volatile of fluorescein; \(\gamma\) is a constant, represents the renewal rate of fluorescein; \(f(x_i(t + 1))\) is the fitness function values.

Neighbours in the collection, when fireflies \(i\) find higher luciferin value of firefly \(j\), if the firefly \(i\) and fireflies \(j\) distance less than the radius of decision-making, the firefly \(i\) will be in \(p_{ij}(t)\) (calculated by formula (3)) probability moved to the direction of the firefly \(j\); Then according to formula (4) position updates and compute the objective function value for the new position; and finally, to update luciferin value according to equation (5).

3. BP neural network

BP neural network is a multi-layer feedforward neural network, with the input layer, hidden layer and output layer. Literature (Wang & Shi, 2002) is given the number of input layer neurons is \(m\), the number of output layer neurons is \(1\), choose the number of hidden layer neurons \(p\)-layer BP neural network, its network structure shown in
Figure 1, the neural network mapping is complete with \( f : \mathbb{R}^m \rightarrow \mathbb{R}^1 \), which was expressed as formula (6) below:

\[
x_{i+1} = \frac{1}{1 + \exp \left( - \sum_{j=1}^{p} c_j b_j + \varepsilon \right)}
\]

In the formula, \( c_j \) is the hidden layer to the output layer connection weights, \( b_j \) is the output node of the hidden layer, \( \varepsilon \) is the threshold value of the output layer.

The BP neural network uses the Sigmoid function \( f(x) = \frac{1}{1 + e^{-x}} \) as the transfer function, and the output of the hidden layer nodes is shown in the formula (7):

\[
b_j = \frac{1}{1 + \exp \left( - \frac{m}{\tau} \sum_{i=1}^{n} w_{ij} x_i + \theta_j \right)}, \quad j = 1, 2, \ldots, p
\]

In the formula, \( w_{ij} \) is the weights between input layer to hidden layer, \( \theta_j \) is the threshold of hidden layer nodes. The BP neural network threshold of \( \theta_j, \varepsilon \) and connection weights \( w_{ij} \) can be obtained through the BP neural network training, so \( x_{i+1} \) can be predicted.

Gradient descent method is the learning rule of BP network. That is to say, the weight of the network is modified according to the direction of negative gradient. After correction, the error objective function is minimized. The objective function is shown in Equation (8):

\[
E_p = \frac{1}{2} \sum_{j=1}^{N} (t_{pj} - o_{pj})^2
\]

In the formula, \( E_p \) is the error function of the sample of \( p \)th, \( t_{pj} \) is the \( j \)th component of the \( p \)th output sample, \( o_{pj} \) is the \( j \)th network output value of the \( p \)th sample.

Using gradient descent method to train BP network often makes the neural network fall into local minimum. Reducing network performance, therefore, this paper uses firefly algorithm to train BP neural network.

4. Reconstruction of phase space

Phase space reconstruction theory is the basis of chaotic time series prediction, phase space reconstruction through one-dimensional time series construct the original system of phase space structure reversely. Packard and Takens proposed a method to reconstruct the phase space of coordinate delay by using the time delay sequence (Packard, Crutchfield, Farmer, & Shaw, 1980; Takens, 1981), The essence of the coordinate delay method is to construct the phase space vector of \( m \)-dimension by the delay of different time series of chaotic time series:

\[
X_i = (x_i, x_{i+\tau}, \ldots, x_{i+(m-1)\tau})^T (i = 1, 2, \ldots, M)
\]

In the formula, \( M = n-(m-1)\tau \) is the number of the reconstruction space; \( \tau \) is the delay time; \( m \) is the embedded dimension, that is the reconstruction phase space dimension.

Therefore, for the prediction signal of \( n \) data points \( x_1, x_2, x_3, \ldots, x_n \), can be in the \( m \) dimensional phase space to form a \( M = n-(m-1)\tau \) state point. These phase points connections form the trajectory of \( n \) data points in \( m \)-dimensional phase space, which characterize the evolution of the system status over time.

Takens theorem proved that if the embedding dimension \( m \geq 2d + 1 \), \( d \) is dimension of system dynamics, the phase space constituted by the original system state variables is equivalent with the dynamic behaviour one-dimensional measurements in the reconstructed phase space, two-phase space of chaotic attractor points diffeomorphism, that is Takes Theorem guarantees that we
can reconstruct the phase space with the original power system in the topological sense equivalence from one-dimensional chaotic time series, this provides a basis for predicting chaotic time series. Coordinate delay phase space reconstruction technology has two key parameters, namely embedding determine \( m \) and the determine of delay time \( \tau \).

### 4.1. Calculation of embedding dimension \( m \)

In Taken’s theorem, for the ideal of infinite and noiseless one-dimensional time series, embedding dimension \( m \) and the delay time \( \tau \) can take any value, but in fact the time series are finite length sequences containing noise, embedding dimension and time delay cannot be any value. Otherwise, it will affect the quality of phase space reconstruction. The purpose of determining the embedding dimension is to make the original attractor and the reconstructed attractor topological equivalent. Cao method is used to calculate the embedding dimension (Cheng & Ko, 2006), for the time series \( \{x(n)\} \),

\[
a(i, d) = \frac{||x_{d+1}(i) - x_{dN}^N(i)||}{||x_d(i) - x_{dN}^N(i)||} \tag{10}
\]

where \( x_d(i) \) and \( x_{dN}^N(i) \) are the \( i \) vector of \( d \) dimension space and its closest point. \( x_{d+1}(i) \) and \( x_{dN}^N(i) \) are the \( i \) vector of \( d + 1 \) dimension space and its closest point. The average value of \( a(i,d) \) is calculated.

\[
E(m) = \frac{1}{N - m \tau} \sum_{i=1}^{N-m \tau} a(i,m) \tag{11}
\]

Then seek \( E_1(m) \).

\[
E_1(m) = \frac{E(m + 1)}{E(m)} \tag{12}
\]

If the time series are determined, then \( E_1(m) \) will not change with the increase of \( m \) to a certain value, and the embedding dimension \( m \) is determined.

### 4.2. Calculation of delay time \( \tau \)

At present, the main method for determining the delay time \( \tau \) is the sequence correlation method and the phase space geometry method. Auto-correlation method is a kind of serial correlation method, although the auto-correlation function is a simple method to calculate the delay time, but it can only extract the linear correlation between time series. The average displacement method belongs to the phase space reconstruction method, which can be linked to the correlation criterion, but the method may have a strong shake in the overall variation trend, which has a certain randomness. Therefore, this paper uses the multiple correlation method combined the auto-correlation method with the average displacement method (Wang, Sun, Fei, & Zhu, 2007), which has little computation complexity, has a strong anti-noise ability, specific calculation is as follows:

The average displacement of the time series \( \{x(n)\} \) which under the reconstruction of \( m \)-dimensional phase space is \( \langle S^2_m(\tau) \rangle \), and the expression is like the formula (12):

\[
\langle S^2_m(\tau) \rangle = \frac{1}{N} \sum_{i=0}^{N-1} \sum_{j=1}^{m-1} (x_{i+j\tau} - x_i)^2 \tag{13}
\]

where \( N \) is points of the observation sequence, ignoring the edge point difference, \( \sum_{i=0}^{N-1} x_i^2 \) is recorded as a constant in the range of \([0, m–1] \), denoted as \( E = \frac{1}{N} \sum_{i=0}^{N-1} x_i^2 \), substituted into the formula (12) can be obtained:

\[
\langle S^2_m(\tau) \rangle = 2(m - 1)E - 2 \sum_{j=1}^{m-1} R_{xx}(j\tau) \tag{14}
\]

Among them, \( R_{xx}(j\tau) \) is the auto-correlation function of the sequence at the time span of \( j\tau \), that is

\[
R_{xx}(j\tau) = \frac{1}{N} \sum_{i=0}^{N-1} x_i x_{i+j\tau}.
\]

The complex correlation function defined by the formula (13) is:

\[
R_{xx}^m(\tau) = \sum_{j=1}^{m-1} R_{xx}(j\tau) \tag{15}
\]

The first zero of the (14) type is the value of the time delay \( \tau \).

### 5. The design of the Glowworm Swarm Algorithm to optimize the BP neural network algorithm

#### 5.1. Basic thought

The basic idea of GSOBPNN is: According to the input and output parameters to determine the network structure of the BPNN, which determine the Glowworm Swarm Algorithm coding length of each individual. Population of each individual contains the ownership of the BPNN value and threshold value, individual fitness function value calculated by the fitness function, and through the location update, decision-making radius update and fluorescein update to find the best fitness function value corresponding to the individual. The BPNN initial weights and threshold values are assigned by the best individual which obtained by the Glowworm Swarm Algorithm.
optimization, reuse BPNN network optimization model to obtain the predicted value BPNN which have global optimal solution.

5.2. Optimization of BP neural network algorithm based on Glowworm Swarm Algorithm

The basic steps of the algorithm are as follows.

Step 1: coding. Individual coding method of Glowworm Swarm Algorithm real number coding, each individual is represented by a real number string, the real number is composed of 4 parts, namely: the connection weights between the hidden layer and the output layer, the connection weights between the input layer and the hidden layer, the hidden layer, the hidden layer threshold and the output layer threshold. Each individual contains all the weight and threshold value of BPNN, each individual can represent a structure of BPNN.

Step 2: Initialization algorithm parameter. Randomly generated individual species of firefly \(n\) to form the initial firefly population, set up an initial fluorescein \(L_0\) of each firefly, the perception radius \(R_s\), the initial step length \(s\), the maximum step size \(s_{\text{max}}\), the smallest step size \(s_{\text{min}}\), fluorescein evaporation rate \(\rho\), fluorescein update rate \(\gamma\), the initial value of the iteration counter \(t = 0\), set the maximum number of iterations of the algorithm \(t_{\text{max}}\).

Step 3: Calculation of fitness function value. Fitness is the main index to describe the quality of the individual. This paper selects (Normal Root Mean Square Error, NRMSE) as the fitness function. Its expression as the formula (15):

\[
f = \left\{ \frac{1}{(S - 1)\sigma^2} \sum_{t=1}^{S} (y_p - y_p)^2 \right\}^{1/2}
\]  

In the formula, \(S\) is the number of predicting samples, represents the standard deviation of the time series, \(t_p\) is the desired output of the samples \(p\), \(y_p\) is the actual output of the samples.

Step 4: Fluorescein update. According to the formula (5) to update each firefly luciferase value in generation \(t\).

Step 5: Position update. In the Glowworm Swarm Algorithm, when the firefly \(i\) find firefly \(j\) which has higher fluorescein on it, if the distance between the two is less than the radius of decision, the firefly \(i\) will according to the probability of the formula (3) to be near the firefly \(j\); Then, use the formula (5) to calculate each individual fireflies fluorescence factor, and the position of the object is updated according to the formula (4) and the value of the target function is calculated by the formula (15), then the global optimal value is updated.

Step 6: Decision radius updated. According to formula (1) update dynamic decision radius.

Step 7: If the number of iterations exceeds the maximum evolution times, or achieve accuracy requirements set by the user, then quit the operation, otherwise returns to Step 4 execution.

Step 8: Decomposing the individual which obtained by the Glowworm Swarm Algorithm optimization into the connection weight and threshold of BPNN, which can be used as the initial weights and thresholds of the forecasting model, and the optimal solution of the network traffic prediction is obtained by training BPNN forecasting network.

6. Simulation experiment

6.1. The GSOBPNN model prediction

In this paper, the SNMP protocol data acquisition method is used to monitor the flow of a CISCO6509 device to a port in Hubei information centre related nodes. The monitoring time is from 15–18 August 2015, and the sampling time interval is 5 min. Taking 500 of the data, the 500 data is divided into two parts, the first 400 data as a training set, as the training sample, the 100 data as a test sample, Table 1 gives the test set of data. Separately with BPNN and GSOBPNN two prediction models to forecast the data set, Figure 2 shows the predicted results of BPNN prediction model, in order to facilitate comparison, Figure 3 shows the forecast result which in the same scale of GSOBPNN and IGSOBPNN prediction model.

From Figures 1 and 2, we can see that the prediction results of the two kinds of prediction model can well predict the tendency of changes in the measured network traffic, and the GSOBP prediction results of neural network forecasting model is superior to the BP neural network prediction model, which shows that the Glowworm Swarm Algorithm to optimize the BP neural network for network traffic prediction is feasible and effective.

6.2. Predictive performance analysis of different embedding dimensions and time delays

Five hundred sets of experimental data collected by a centre of information in Hubei Province are regarded as a set of time series data. The maximum Lyapunov exponent of the time series is calculated by WOLF method. It is calculated that \(\lambda_1 = 0.0392 > 0\). Therefore, the network traffic time series in this paper can be regarded as chaotic time series. Then the method of phase space reconstruction is used to deal with chaotic time series. Mutual information method and Cao method are used to determine the time delay and embedding dimension. The optimal time delay is 2 and the optimal embedding dimension is 3.
Table 1. Basic data for test sets.

| No. | Traffic/M bit | No. | Traffic/M bit | No. | Traffic/M bit | No. | Traffic/M bit |
|-----|--------------|-----|--------------|-----|--------------|-----|--------------|
| 1   | 164.59       | 26  | 215.92       | 51  | 289.27       | 76  | 222.67       |
| 2   | 146.7        | 27  | 253.44       | 52  | 293.53       | 77  | 255.93       |
| 3   | 72.09        | 28  | 247.99       | 53  | 286.4        | 78  | 272.07       |
| 4   | 178.93       | 29  | 192.07       | 54  | 305.53       | 80  | 247.47       |
| 5   | 89.46        | 30  | 256.53       | 55  | 279.07       | 81  | 235.67       |
| 6   | 96.84        | 31  | 261          | 56  | 289.4        | 82  | 229.73       |
| 7   | 127.73       | 32  | 196.8        | 57  | 279.13       | 83  | 223.73       |
| 8   | 165.59       | 33  | 192.07       | 58  | 304.27       | 84  | 225.47       |
| 9   | 149.43       | 34  | 231.16       | 59  | 287.4        | 85  | 196.49       |
| 10  | 128.35       | 35  | 222.67       | 60  | 258.6        | 86  | 164.49       |
| 11  | 137.9        | 36  | 273.47       | 61  | 191.13       | 87  | 161.93       |
| 12  | 129.01       | 37  | 256.93       | 62  | 255.27       | 88  | 170.71       |
| 13  | 127.92       | 38  | 321.2        | 63  | 255.27       | 89  | 171.62       |
| 14  | 154.89       | 39  | 339.47       | 64  | 272.93       | 90  | 164.49       |
| 15  | 145.6        | 40  | 345.27       | 65  | 297.27       | 91  | 161.93       |
| 16  | 126.11       | 41  | 369.8        | 66  | 278.2        | 92  | 170.71       |
| 17  | 92.78        | 42  | 352.93       | 67  | 275.6        | 93  | 168.21       |
| 18  | 133.32       | 43  | 263          | 68  | 276.4        | 94  | 163.28       |
| 19  | 139.07       | 44  | 238.2        | 69  | 288.2        | 95  | 186.31       |
| 20  | 141.19       | 45  | 246.33       | 70  | 295.8        | 96  | 208.49       |
| 21  | 139.56       | 46  | 272.93       | 71  | 269.73       | 97  | 323.8        |
| 22  | 135.79       | 47  | 274.53       | 72  | 255.27       | 98  | 311.13       |
| 23  | 140.57       | 48  | 259.27       | 73  | 185.07       | 99  | 273.47       |
| 24  | 155.66       | 49  | 258.47       | 74  | 204.27       | 100 | 207.73       |

Figure 2. The BPNN model prediction results and prediction error.

Due to the different network structure, the prediction results are also different. Embedding dimension and time delay determine the input and output of the structure. Different embedding dimensions and time delays are selected to continue the prediction experiment. When $m = 3$, $\tau = 1$, the input is $X(t) = [x(t - 2), x(t - 1), x(t)]$, output is $s(t) = x(t + 1)$. When $m = 4$, $\tau = 1$, the input is $X(t) = [x(t - 3), x(t - 2), x(t - 1), x(t)]$, output is $s(t) = x(t + 1)$. The results of all prediction models are summarized in Table 2. Regularized root mean square error (RMSE) is used as the evaluation index of prediction results.

As can be seen from Table 2, the table reflects the prediction results of BPNN network prediction model and GSOBPNN network prediction model under different embedding dimensions and time delays. GSOBPNN

Figure 3. The GSOBPNN model prediction results and prediction error.
method has a high prediction effect both in the optimal embedding dimension time delay and in the non-optimal embedding dimension time delay. So GSOBPNN method is effective in network traffic time series prediction. Well, As can be seen from Table 2, Either method has better prediction effect under the optimal embedding dimension time delay than under the non-optimal embedding dimension time delay. This shows that the embedding dimension and time delay are also the main factors affecting the prediction results.

7. Conclusion

Network traffic prediction has the characteristics of nonlinear, long correlation, time varying, self-similarity and sudden characteristics, and it is difficult to get the ideal result by using the traditional linear prediction model. Based on nonlinear time series, this paper uses the improved differential evolution algorithm to optimize the BP neural network centre \( c_i \), the width \( \sigma_i \), the network connection weights \( w_{kj} \) of the hidden unit basis function, so as to better play the BP neural network and the fitting effect of nonlinear approximation capability and get the best neural network prediction model. The GSOBPNN forecasting model is applied to forecast the network traffic, and the accuracy of prediction is compared with that of BPNN. The simulation results show that the proposed method can be used to predict the network traffic more accurately and has good application prospect in network traffic prediction.

Disclosure statement

No potential conflict of interest was reported by the author.

Funding

This study is supported by the Science and Technology Research Projects in 2018 of Henan Province, China [No. 182102410064].

References

Atiya, A. F., Alym, A., & Parlosa, G. (2005). Sparse basis selection: New results and application to adaptive prediction of video source traffic. IEEE Trans on Neural Networks, 16(5), 1136–1146.

Beran, J., Sherman, R., & Taqqu, M. S. (1995). Long-range dependence in variable-bit-rate video traffic. IEEE Trans on Communications, 43(2), 1566–1579.

Cheng, M. Y., & Ko, C. H. (2006). A genetic – fuzzy-neuro model encodes FNNs using SWRM and BRM. Engineering Applications of Artificial Intelligence, 19(8), 891–903.

Grunenfelder, R., Cosmas, J. P., Manthorpe, S., & Odinma-Okafor, A. (1991). Characterization of video codes as autoregressive moving average processes and related queueing system performance. IEEE Journal of Selected Areas in Communications, 9(3), 284–293.

Heffes, H., & Lucantoni, D. (1986). A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance. IEEE Journal of Selected Areas in Communications, 4(6), 856–868.

Heyman, D. P., Tabatabai, A., & Lakshman, T. V. (1992). Statistical analysis and simulation study of video teleconference traffic in ATM networks. IEEE Transactions on Circuits and Systems for Video Technology, 2(1), 49–59.

Hotnik, K., Stinchcombe, M., & White, H. (1989). Multilayer feed forward networks are universal approximators. Neural Networks, 1(2), 3592366.

Hou, Y., & Wu, Z. D. (2015). Network traffic prediction based on improved T-S fuzzy neural network. Journal of University of Jinan, 29(4), 280–285.

Hussein, D. (2001). An object-oriented neural network approach to short-term traffic forecasting. European Journal of Operational Research, 131(2), 253–261.

Krishhand, K. N., & Ghose, D. (2009). Glowworm Swarm optimization for simultaneous capture of multiple local optimal of multimodal functions. Swarm Intelligence, 3(2), 87–124.

Li, H. Y., & Li, D. (2011). Study on network traffic forecasting based on improved neural network. Computer Simulation, 28(9), 132–135.

Maglaris, B., Anastassiou, D., Sen, P., Karlsson, G., & Robbins, J. D. (1988). Performance models of statistical multiplexing in packet video communications. IEEE Transactions on Communications, 36(7), 834–844.

Packard, N. H., Crutchfield, J. P., Farmer, J. D., & Shaw, R. S. (1980). Shaw geometry from a time series. Physical Letters, 45(9), 712–716.

Park, K., & Willinger, W. (2000). Self-Similar network traffic and performance evaluation (Vol. 38, No. 5, pp. 1–38). New York, NY: Wiley.

Roughan, M., Veitch, D., & Abyr, P. (2000). Real-time estimation of the parameters of long-range dependence. IEEE/ACM Trans on Networking (TON), 8(4), 467–479.

Takens, F. (1981). Detecting strange attractors in turbulence. Lecture notes in mathematics (No. 898, pp. 361–381).

Wang, H. Y., & Shi, G. D. (2002). The technology of artificial neural network and application (pp. 34–36). Beijing: Sinopec Press.

Wang, J., Sun, L., Fei, X. Y., & Zhu, B. (2007). Chaos analysis of the electrical signal time series evoked by acupuncture. Chaos, Solitons & Fractals, 33(3), 901–907.

Yu, H. (2013). Network traffic prediction based on wavelet analysis and Hopfield neural network. Computer Applications & Software, 30(6), 246–249.

Zhao, Q. Y. (2013). Network traffic prediction based on neural network optimized by GA. Microelectronics & Computer, 30(3), 32–39.

Table 2. Prediction errors based on different embedding dimensions and time delays.

| Prediction method | m = 4, r = 1 | m = 3, r = 1 | m = 3, r = 2 |
|-------------------|-------------|-------------|-------------|
| BPNN              | 0.3912      | 0.3324      | 0.2791      |
| GSOBPNN           | 0.2572      | 0.2356      | 0.2218      |