TELKOMNIKA, Vol.11, No.3, March 2013, pp. 1683 ~ 1690
ISSN: 2302-4046

Chaotic Prediction for Traffic Flow of Improved BP Neural Network

Yue Hou*[1], Yuemei Mai[2]
[1,2]School of Electronic and Information Engineering, Lan Zhou Jiao Tong University,
Lan Zhou 730070, P.R.China
Corresponding author, e-mail: houyue@mail.lzjtu.cn[1], yuemeimai@163.com[2]

Abstract

A prediction algorithm for traffic flow prediction of BP neural based on Differential Evolution (DE) is proposed to overcome the problems such as long computing time and easy to fall into local minimum by combing DE and neural network. In the algorithm, DE is used to optimize the thresholds and weights of BP neural network, and the BP neural network is used to search for the optimal solution. The efficiency of the proposed prediction method is tested by the simulation of two typical chaotic time series and real traffic flow. The simulation results show that the proposed method has higher precision compared with the traditional BP neural network, so prove it is feasible and effective in the practical prediction of traffic flow.

Keywords*: prediction; traffic flow; BP neural network; differential evolution (DE)*

1. Introduction

Traffic guidance and control is the important part of the intelligent traffic system. Real-time and precise traffic flow is the premise and the key to the realization of traffic guidance and control [1, 2]. Urban traffic flow system has significant chaotic characteristic, and its short-term traffic flow data is the chaotic time series. The thought, based on that fact, is to build a non-linear mapping, which is to build a prediction model to restore its original system approximately. So far, many scholars have made a lot of research in this field and built various traffic flow prediction models such as Volterra filter adaptive model [3], BP neural network model [4] and RBF neural network model [5]. Among these models, neural network becomes the hot spot because of its great learning power and good generalization ability. However, values of thresholds and weights of neural network have a greater influence to the performance in the practical use [6]. Differential Evolution (DE), proposed in 1995 by Storn, is an algorithm based on group optimization [7]. The algorithm has great global searching ability and simple principle with fewer controlled parameters to realize easily, so it's very adaptable for neural network parameters optimization.

From the perspective of non-linear time series, this article, using chaotic dynamics theory to analyze the short-term traffic flow, puts forward a BP neural network method based on DE (DEBP). When apply this method to two typical chaotic time series and traffic flow time series, results show that the method has greater non-linear fitting ability and higher predicating accuracy.

2. Basic Differential Evolution Algorithm

Differential evolution (DE) algorithm is an improved algorithm based on group evolution with characteristics of optimal value of memorial individual and group-in information sharing, which is to optimize the value of the problem using cooperation and competition among individuals in the group [7].

First, get a group of random initialed population:

$$X^0 = [x_1^0, x_2^0, .., x_{NP}^0] \tag{1}$$

Received November 16, 2012; Revised January 26, 2013; Accepted February 9, 2013

NP is the size of the population and D is the dimension. Under a series of operation, evolution of individuals of the t generation is $x_i^t = [x_{i,1}^t, x_{i,2}^t, ..., x_{i,D}^t]$ .The principle of algorithm is that the difference vector obtained by the division between two random different individuals of the parent generation added on a random selected individual creates a mutation individual. Then according to certain probability, crossover parent individuals and mutation individuals to create a new individual, which compares with parent individuals according to the value of sufficiency function, and then select individuals with the optimal sufficiency as child generation.

2.1. Mutation Operation

Mutation operation can avoid evolution falling into partial optimal value. The basic mutation part of DE is the difference vector of parent generation and each vector pair contains two different individuals (x_{r1}^t, x_{r2}^t) . With the fact that mutation individuals have different creating modes, varieties evolution scenarios are formed and the basic mutation is equation (2).

$$x_m = x_{r3}^t + F * (x_{r1}^t - x_{r2}^t) \qquad (2)$$

Where x_{r1}^t and x_{r2}^t are different random individuals and $F \in [0,2]$ is the zoom factor.

2.2. Crossover Operation

DE uses crossover operation to maintain group diversity. Crossover strategy is: making crossover operation between the *i* individual x_i^t and x_m of the group to create a test individual x_T .To guarantee individual evolution, first through random selection, assuring x_T is provided by at least one x_m and other bits are obtained by CR. The function of crossover operation is equation (3).

$$x_{Tj} = \begin{cases} x_{mj} & rand() \le CR \\ x_{ij}^t & rand() > CR \end{cases} \quad j = 1,2,...,D \qquad (3)$$

where $rand()$ is the random value in [0, 1]; $CR \in$ [0,1]. The bigger CR is, more benefit for accelerating convergence speed. The smaller CR is, more benefit for maintaining group diversity and global search.

2.3. Selection Operation.

DE adopts greedy searching strategy to select child generation with high sufficiency, and the function of selection operation is equation (4).

$$x_i^{t+1} = \begin{cases} x_T & f(x_T) < f(x_i^t) \\ x_i^t & f(x_T) \ge f(x_i^t) \end{cases} \qquad (4)$$

3. BP Neural Network

BP neural network is a kind of multilayer forward network which contains input layer, output layer and implication layer. The document presents a typical 3-layer BP neural network [8], and the quantity of neuron of output layer with better predictive effect in chaotic time series prediction is *m*, the quantity of neuron of selected implication layer is *p*, the quantity of output layer is 1 so mapping of neural network is $f : R^m \to R^1$ and its expression is equation (5).

$$x_{i+1} = f(X_i) = \frac{1}{1+\exp(-\sum_{j=1}^{p} c_j b_j + \varepsilon)} \tag{5}$$

Where c_j the link weight from implication is layer to output layer; ε is the threshold of output layer and b_j is the output of nodes of implication layer.

Transition function of BP neural network is sigmoid function $f(x) = \frac{1}{1+e^{-x}}$, thus we have equation (6).

$$b_j = \frac{1}{1+\exp(-\sum_{i=1}^{m} w_{ij} x_i + \theta_j)}, \quad j = 1,2,\cdots,p \tag{6}$$

where w_{ij} is the link weight from input layer to implication layer; θ_j is the threshold of nodes of implication layer; link weight w_{ij}, c_j and threshold θ_j, ε can obtained by BP neural network training so x_{i+1} is predicable. Equation 4 is prediction model of chaotic time series of BP neural network, normally *p* is $2m+1$.

4. Algorithm Design of BP Neural Network Based on DE

4.1. Principle

The principle of DEBP is: list the possible existing neurons in neural network and make the possible link weights and thresholds of these neurons before training binary code string or individuals represented by real code string, furthermore random generate population of these strings and enhance the population diversity using random selection, mutation and crossover. Through mutation and crossover operation, a new temporary population is created. Using strategy to make optimized selection of individuals of population, thereby a new population is created once again. In accordance with the process optimal individual would be found. Assign the optimal individual obtained by DE to initial weight and threshold, and then use BP neural network prediction model to find the best to get prediction value of BP neural network with global optimal value.

4.2. Algorithm of BP Neural Network Based on DE.

Steps of the algorithm are:

(i) Code: DE uses real code which the length of individual code is equal to the number of its variable. This paper codes BP neural network's parameters w_{ij}, c_j, θ_j and ε together into one individual and each individual represents a BP network structure.

(ii) Initial population and parameters of algorithm: given the size of population as *NP*, the initial population random created *NP* individuals as $W = (W_1, W_2, \cdots, W_p)^T$, zoom factor as *M* and initial value of crossover probability factor *CR*. Setting the biggest iteration of algorithm *k*, get a real vector $w_1, w_2, \cdots, w_t$ of individual W_i as a chromosome of DE.

(iii) Sufficiency function: sufficiency is the main index describing merit degree of individuals in population in DE. This paper, we adopt mean square error as sufficiency function and the expression is equation (7).

$$G = \frac{1}{N}\sum_{p=1}^{N}(t_p - y_p)^2 \tag{7}$$

Where *N* is the total sum of training samples; t_p is the expectation value of the *p* sample; y_p is the actual output of the *p* sample. Calculate the sufficiency of each individual and reserve the individual with the minimum sufficiency.

(iv) Mutation: make mutation operation to individual W_i according to equation (2) to create mutation individual $W_i^{'}$.

(v) Crossover: make crossover operation to mutation individuals W_i and $W_i^{'}$ according to equation (3), and then create new individual W_T.

(vi) Selection: substitute and into object function and according to equation (4) select individual with small value of sufficiency function $W_T^{'}$ as individuals of new population.

Separate the individual obtained by DE into the link weight and threshold of BP neural network which also worked as the initial weight and threshold of prediction model. Make training of BP neural network prediction network to get the optimal value of chaotic time series prediction.

5. Simulation Experiment

Apply DEBP prediction model to prediction of real traffic flow time series then compare with BP neural network prediction model to confirm the validity of the method.

5.1. Prediction Assessment Standards

Error assessment of experiment mainly uses RMSE, NRMSE and Re, which are expressed in equation (8) to (11).

$$RMSE = \left\{\frac{1}{S-1}\sum_{t=1}^{S}\left[\hat{y}(t) - y(t)\right]^2\right\}^{1/2} \tag{8}$$

$$NRMSE = \left\{\frac{1}{(S-1)\sigma^2}\sum_{t=1}^{S}\left[\hat{y}(t) - y(t)\right]^2\right\}^{1/2} \tag{9}$$

$$\mathrm{Re} = \frac{\sum_{t=1}^{S}\left[\hat{y}(t) - y(t)\right]^2}{\sum_{t=1}^{S} y^2(t)} \tag{10}$$

Where S the number of prediction is samples; $y'(t)$ and $y(t)$ are separately prediction and expectation value; σ represents standards variance of object time series. Apply follow expression to make normalization to time series data of experiment and then make phase space reconstruction [9].

$$x_i' = \frac{x_i - \frac{1}{n}\sum_{i=1}^{n} x_i}{\max(x_i) - \min(x_i)} \tag{11}$$

Where $\{x_i\}$ is original time series and $\{x_i'\}$ is normalization time series.

5.2. Two Typical Chaotic Time Series

Henon chaotic time series:

The mathematical model of chaotic time series is equation (12).

$$\begin{cases} x(k+1) = 1 + y(k) - ax^2(k) \\ y(k+1) = bx(k) \end{cases} \tag{12}$$

When, $a = 1.4$, $b = 0.3$, system is in chaotic condition, the power system through phase space reconstruction after the iteration is shown in Figure 1.

Lorenz chaotic time series prediction.

The mathematical model of chaotic time series is equation (13).

$$\begin{cases} \dfrac{dx}{dt} = a(y - x) \\ \dfrac{dy}{dt} = (c - z)x - y \\ \dfrac{dz}{dt} = xy - bz \end{cases} \tag{13}$$

When $a = 10$, $b = 8/3$, $c = 28$, system is in chaotic condition, three dimensional phase space trajectory and the two-dimensional phase plane attractor through phase space reconstruction are shown in Figure 2.

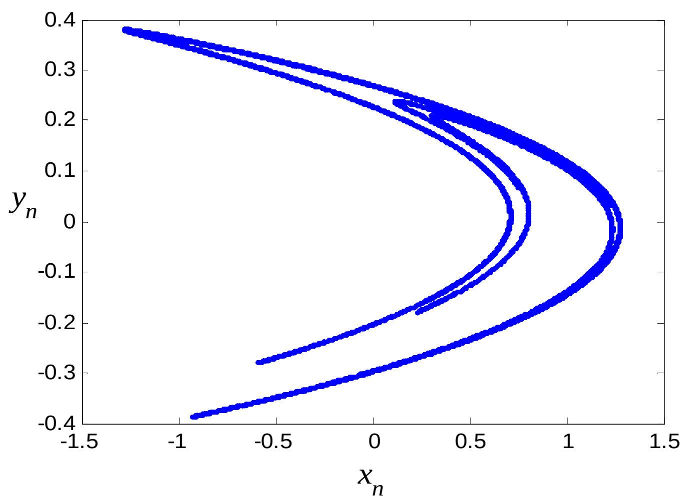


Figure 1. Henon system iterative of a=1.4 and b=0.3

In experiment, BP neural network structure selects m-5-1 typical three-layer structure, the step size of integral time of two typical chaotic time series is 0.1, embedded dimension m is 4, 4, 2 separately and delay time τ is 1. The number of training is 10,000, training object error is 0.01 and learning rate is 0.1. DE parameters are: set the size of population as 10, the number of evolution generation as 100, crossover rate as 0.4 and mutation rate as 0.2. Take former 1,500 data of chaotic time series as training samples and later

500 data as prediction test samples. Figure 3-Figure 4 shows single-step prediction impression drawing of Lorenz system and Table 1 shows prediction error of two typical chaotic time series.

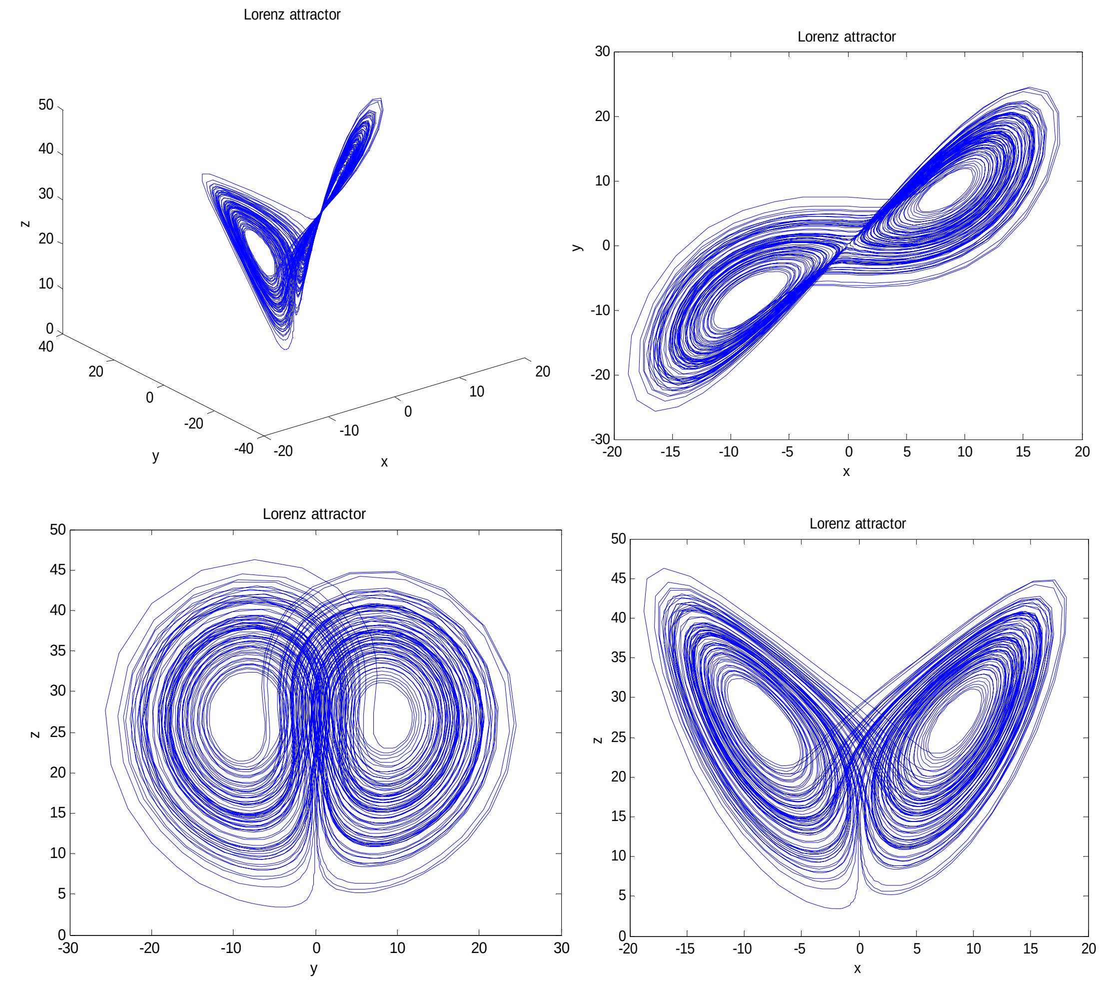
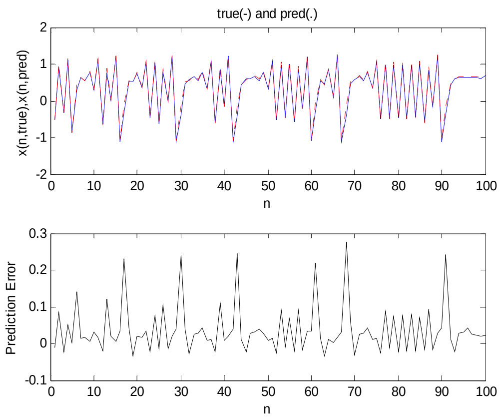
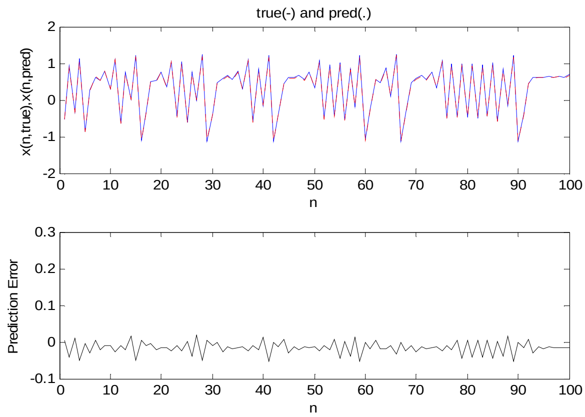


Figure 2. Lorenz attractors through phase space reconstruction

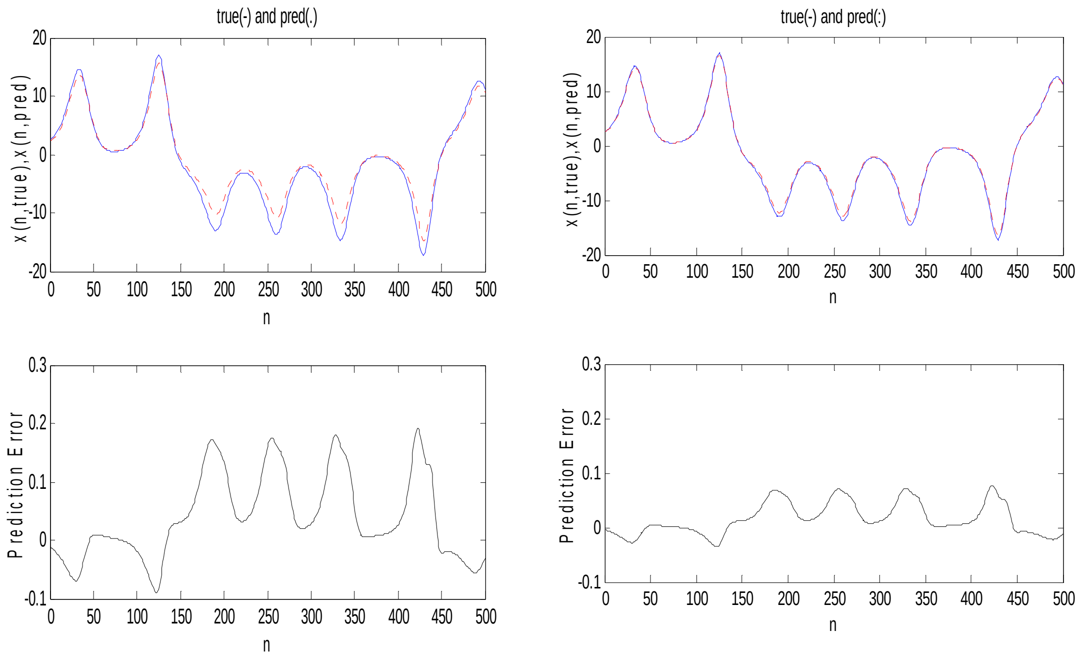


a. The forecasting result based on BP neural network model



b. The forecasting result based on DEBP neural network model

Figure 3. The forecasting result based on Henon chaotic time series



a.The forecasting result based on BP neural network model b.The forecasting result based on DEBP meural network model

Figure 4. The forecasting result based on Lorenz chaotic time series

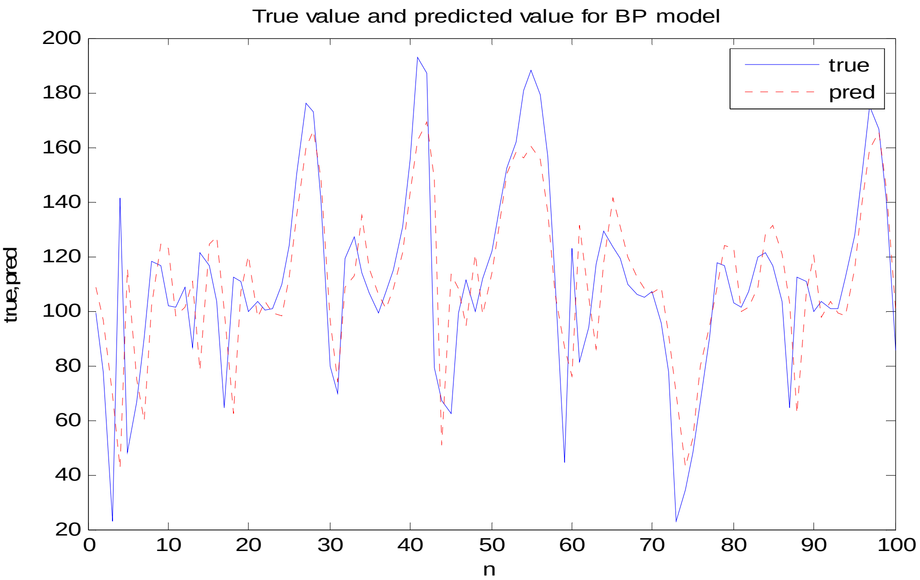
Table 1. Forecasting errors of two typical nonlinear systems

system		Henon	Lorenz
training steps	BP	4580	560
	DEBP	8	2
RMSE	BP	0.1019	0.1038
	DEBP	0.0512	0.0233
NRMSE	BP	0.1380	0.2250
	DEBP	0.0778	0.0504
Re	BP	0.0193	0.0585
	DEBP	0.0049	0.0025

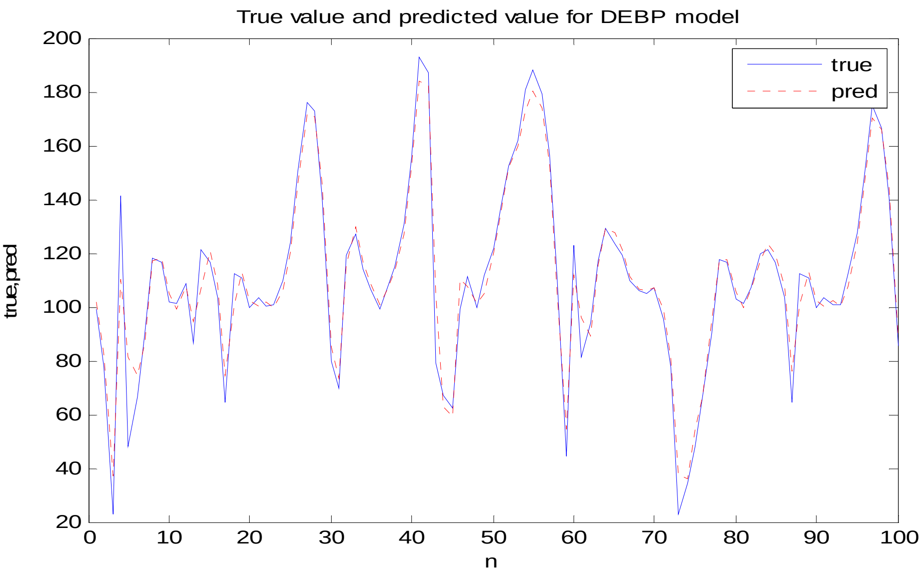
5.3. Prediction of Real Traffic Flow Time Series

The traffic flow data in the simulation came from the traffic data in March, 2011 of the Britain Transport Bureau. Observation time is 6:00-20:00 and the traffic flow data is recorded and calculated every 15 minutes. In this paper, 5 days' data (336 groups) was taken as the research object and the improved algorithm calculating the maximum Lyapunov index was adopted [10]. The results show that the delay time τ is 1, embedding dimension m is 3 and the maximum Lyapunov index is 0.3754 which illustrate that this time series of traffic flow is the chaotic time series.

In the experiment, the number of network training is 5,000, training object error is 0.01, and learning rate is 0.1 and other parameters remain unchanged. Take the former 236 data of traffic flow series as training samples and the later 100 as testing samples to make prediction using DEBP and BP models. Figure 5 shows the prediction results when τ =1, m=3. Take NRMSE as the evaluation index, Table 2 presents prediction errors of two prediction models under different delay time and embedding dimension. Figure 5 and Table 2 explain that the prediction results of the two models are able to predict the tendency of the change of traffic flow properly, and that the predictable accuracy of optimized DEBP model is higher than BP model which explains that using DEBP prediction model to predict real- traffic flow series is valid. Seen from Table 2, we can conclude that the prediction results reach the best when τ and m valued the best optimal delay time and embedding dimension.



a.The forecasting result based on BP neural network model



b.The forecasting result based on DEBP meural network model

Figure 5. Actual measure the forecast results of traffic flow chaotic sequence

Table 2. Traffic flow forecasting errors based on different delay time and embedding dimension

Method Type	$m=4,\tau=1$	$m=3,\tau=2$	$m=3,\tau=1$
BP	0.8815	0.7799	0.7735
DEBP	0.5920	0.5868	0.4470

Compared with typical chaotic time series prediction, the advanced extent of prediction precision of DEBP model acted on traffic flow time series is less which explains that urban traffic flow system has higher complexity.

6. Conclusion

Aiming at high demand on real time of traffic guidance and control and its indicated non-linear and uncertainty, this paper, started from non-linear time series, provides an improved chaotic time series prediction of BP neural network based on DE and applies the method to the prediction of real traffic flow system, and then makes comparison with BP neural network prediction model on prediction precision. Results show that the new model has better non-linear fitting ability and higher prediction precision on typical chaotic time series and traffic flow.

Acknowledgements

This work was financially supported by the national social science foundation (12CGL004) and Gansu provincial natural science foundation (1112RJZA051).

References

[1] Hu JM. *An Applicable Short-term Traffic Flow Forecasting Method Based on Chaotic Theory*. Proc of IEEE 6th International Conference on Intelligent Transportation Systems. 2003; 10: 608-613.
[2] Smith B. Comparison of parametric and nonparametric models for traffic flow forecasting. *Transportation Research Part C*. 2002; 10(4): 303-321.
[3] Song Li, Lijun Liu, Hailin Guo. Comparative of the Predictive Method of Chaos in Short-term Traffic Flow. *System Engineering*. 2009; 27(9): 60-64 .
[4] Chaojun Dong, Zhiyong Liu. Multi-layer Neural Network Involving Chaos Neurons and Its Application to Traffic-flow Prediction.*Journal of System Simulation*. 2007; 19(10): 101-104.
[5] Yumei Zhang, Shiru Qu, Kaige Wen. A Short-term Traffic Flow Forecasting Method Based on Chaos and RBF Neural network. *System Engineering*. 2007; 25(11): 30-34.
[6] Wanfu Zhu, Shijun Zhao. Optimal design of structure for neural networks based on rough sets. *Computer Engineering and Design*. 2007; 28(17): 4210-4212.
[7] Storn R, Price K. *Minimizing the real functions of the ICEC'96 contest by differential evolution*. Proc of IEEE Int Conf on Evolutionary Computation. Nagoya. 1996: 842-844.
[8] Hongyan Wang, Guodong Shi. The Technology of Artificial Nueral Network and Application. Beijing: Sinopec press. 2002: 34-36.
[9] Takens F. Detecting strange attractors in turbulence.*lecture Notes in Mathematics*. 1981; 898: 361-381.
[10] Song Li, Guoguang He. Identification of Chaos in the Traffic Flow based on the Improved Largest Lyapunov Exponents Algorithm. *Journal of Wuhan University of Technology (Transportation Science & Engineering)*. 2006; 5: 747-750.