Ensemble of Flexible Neural Tree and Ordinary Differential Equations for Small-time Scale Network Traffic Prediction

Bin Yang[1], Mingyan Jiang[1]*
1. School of Information Science and Engineering, Shandong University, Jinan, 250100, P.R. China
Email: batsi@126.com, jiangmingyan@sdu.edu.cn

Yuehui Chen[2], Qingfang Meng[2] and Ajith Abraham[3]
2. School of Information Science and Engineering, University of Jinan, 106 Jiwei Road, Jinan, 250022, P.R. China
3. Machine Intelligence Research Labs (MIR Labs), Scientific Network for Innovation and Research Excellence
P.O. Box 2259 Auburn, Washington, USA

***Abstract*—Accurate models play important roles in capturing the salient characteristics of the network traffic, analyzing and simulating for the network dynamic, and improving the predictive ability for system dynamics. In this study, the ensemble of the flexible neural tree (FNT) and system models expressed by the ordinary differential equations (ODEs) is proposed to further improve the accuracy of time series forecasting. Firstly, the additive tree model is introduced to represent more precisely ODEs for the network dynamics. Secondly, the structures and parameters of FNT and the additive tree model are optimized based on the Genetic Programming (GP) and the Particle Swarm Optimization algorithm (PSO). Finally, the expected level of performance is verified by using the proposed method, which provides a reliable forecast model for small-time scale network traffic. Experimental results reveal that the proposed method is able to estimate the small-time scale network traffic measurement data with decent accuracy.**

***Index Terms*—hybrid evolutionary method, small-time scale network traffic, the additive tree models, ordinary differential equations, ensemble learning**

I. INTRODUCTION

Network traffic modeling and analysis play a major role in characterizing network performance. Accurate models also play important roles in capturing the salient characteristics of the network traffic, analyzing for the network dynamic, and improving the forecasting ability for system dynamics. It has a fundamental meaning for many network designs and engineering problems, e.g., switcher designing, router, the management of devices, and its software development.

Complexity is common feature in network geometry and information traffic. Complexity appears in practical situations of network traffic, such as the long-range correlations and self-similarity found in the statistical analysis of traffic measurements. It is also a convinced evidence of these phenomena at several different time scales. The complexity revealed from the traffic measurements has led to the suggestion that the network traffic cannot be analyzed in the current framework of existing models [1-3]. Other reliable traffic models and tools for quality assessment and control have been developed in [4-9].

Recently, the rapid development of the communication and network technologies results in the uncertain characteristics of the traffic network, especially the nonlinear time series. The nonlinear time series are more accurately predicted by using some models, i.e., neural networks [10-13], support vector machines [14], adaptive algorithms [15-16].

Zhou et al. [17] proposed a new network traffic prediction model based on nonlinear time series ARIMA/GARCH. This model combined linear time series ARIMA model and non-linear GARCH model. A parameters estimation procedure of the proposed ARIMA/GARCH model was provided. Results indicated that this model could capture prominent traffic characteristics and provided better prediction accuracy compared to the existing FARIMA model.

A framework of a traffic prediction model was proposed in [8] for eliminating the noises caused by random travel conditions, where the influence of special factors was calculated quantitatively. This framework combined several artificial intelligence technologies, e.g., wavelet transform, neural network, and fuzzy logic. In addition to developing the prediction framework, a wavelet de-noising method was also delivered and analyzed.

A multiscale decomposition approach to real time traffic prediction was proposed in [9], where the raw traffic data was decomposed into multiple timescales using the wavelet transform. The wavelet coefficients and scaling coefficients at each scale were predicted

*Corresponding author: Mingyan Jiang (jiangmingyan@sdu.edu.cn)

doi:10.4304/jcp.8.12.3039-3046

independently by using the ARIMA model. Then, the predicted wavelet coefficients and scaling coefficient were combined to represent the predicted traffic. The correlation structure of traffic caused by different network mechanisms was captured, which may not be obvious when examining the raw data directly.

Meng et al. [18] applied the nonlinear time series analysis method to small-time scale traffic measurement data. The prediction-based method was used to determine the embedding dimension of the traffic data. Based on the reconstructed phase space, the local support vector machine prediction method was used to predict the traffic measurement data, and the BIC-based neighboring point selection method was used to choose the number of the nearest neighboring points for the local support vector machine regression model.

To predict the time series of internet traffic, a complex network based on genetic programming and particle swarm optimization was proposed in [19]. The structure of complex network was evolved by using genetic programming and the parameters encoded in this structure that were adjusted by using particle swarm optimization algorithm. The results demonstrated that this model had high prediction accuracy and reflected the actual characterize of the real network traffic precisely.

In order to predict the small-time scale traffic measurements data, a hybrid evolutionary method for identifying a system of ordinary differential equations (ODEs) was presented in [20], where the tree-structure based evolution algorithm and particle swarm optimization (PSO) were used to evolve the architecture and the parameters of the additive tree models for the system of ordinary differential equations. In [21], the flexible neural tree (FNT) model was employed to predict the small-time scale traffic measurements data. Based on the pre-defined instruction/operator sets, the FNT model was created and evolved. This framework satisfied the following requirements: input variables selection, over-layer connections and different activation functions for the various nodes are involved. The FNT structure was developed by using the Genetic Programming, and with which the parameters are optimized by the Particle Swarm Optimization algorithm. Also, the experiment results in [20-21] have demonstrated that the FNT and the system of ordinary differential equations are feasible and efficient for forecasting the small-scale traffic measurements data compared with the feed forward neural network model.

In this paper, our research is to investigate the performance analysis of the ensemble of FNT and ODEs for small-time scale network traffic prediction. The additive tree model is introduced so that more precisely ODEs for the network dynamics can be represented. We use a hybrid evolutionary method, in which GP and PSO are employed to evolve the architecture and the parameters of FNT and the additive tree models for system of ordinary differential equation identification, respectively. The proposed method interleaves both optimizations. To improve the prediction accuracies of time series, two ensemble methods are used.

For the traffic data, we analyze the seemingly chaotic behavior of the small-scale traffic measurements data namely the TCP traffic data [20]. The TCP traffic data contains an hour's worth of all wide-area traffic between Digital Equipment Corporation (DEC) and the rest of the world. The data package used in this paper is DEC-Pkt1, and the time stamps have millisecond precision [20, 22]. The traffic data aggregated with time bin 0.1s, which is the arrived package's amount within the 0.1s time interval [21]. We divide the entire data into roughly two equal halves. No special rules are used to select the training set other than ensuring a reasonable representation of the parameter space of the problem domain [23]. The complexity of the training and test data sets for both indices is depicted in Fig. 1.

This paper is organized as follows. In Section II, the details of the proposed method are described. In Section III, the examples are used to examine the effectiveness and veracity of the proposed method. Finally, the conclusions are drawn in Section IV.

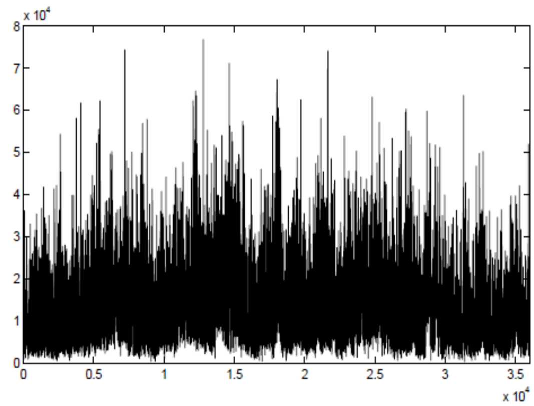

Figure 1. Actual traffic measurements data.

II. Materials and Methods

A. Flexible Neural Tree Model

A FNT model could be viewed as a flexible multi-layer feed forward neural network with over-layer connections and free parameters in activation functions, in which the proper input variables or time-lags for constructing a time-series model could be selected automatically [21]. The adopted function set F and terminal instruction set T for generating a FNT model are described as:

$$S = F \bigcup T = \{+_2, +_3, \ldots, +_N\} \bigcup \{x_1, \ldots, x_n\} \tag{1}$$

where $+_i$ (i = 2, 3,...,N) denotes non-leaf nodes' instructions with i arguments. x_1, x_2, ..., x_n are leaf nodes' instructions and take no other arguments. The output of a non-leaf node is calculated as a flexible neuron model shown in Fig. 2. In this sense, the instruction $+_i$ is also called a flexible neuron operator with i inputs, i real values are randomly generated and used for representing the connection strength between the node $+_i$ and its children [21]. Furthermore, the adjustable parameters a_i and b_i are randomly created as flexible activation function parameters. In this study, the flexible activation function is given by

$$f(a_i, b_i, x) = e^{-(x - a_i / b_i)^2} \tag{2}$$

The output of a flexible neuron $+_n$ is calculated as follows. The total excitation of $+_n$ is

$$net_n = \sum_{j=1}^{n} w_j * x_j \tag{3}$$

where x_j (j = 1,2,...,n) are the inputs to node $+_n$. The output of the node $+_n$ is then calculated by

$$out_n = f(a_n, b_n, net_n) = e^{-(\frac{net_n - a_n}{b_n})^2} \tag{4}$$

A typical flexible neural tree model is shown as Fig. 2. Then, the overall outputs of flexible neural tree are computed from left to right by depth-first method, recursively [21].

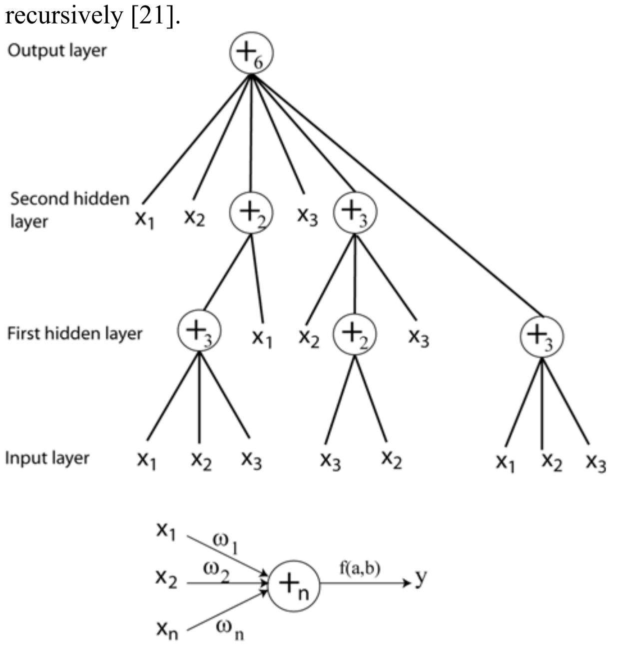

Figure 2. A flexible neuron operator (top), a typical representation of neural tree with function instruction set F = {$+_2$,$+_3$,$+_4$,$+_5$,$+_6$}, and terminal instruction set T = {$x1$, $x2$, $x3$} (bottom).

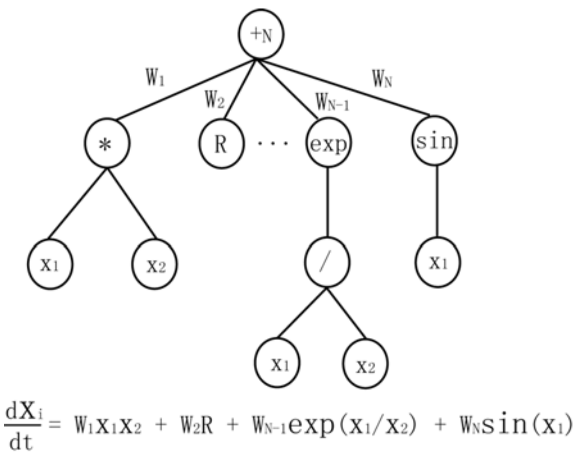

Figure 3. Example of ODEs in the form of the additive tree model.

B. Additive Tree Model

The tree-structure based evolution algorithm is introduced to evolve the architecture of the additive tree models for the system models with expressed by the ODEs. As shown in Fig. 3, the right-hand side of an ODE is encoded into an additive tree individual. Two instruction/operator sets I_0 and I_1 are defined to generate the additive tree in this approach:

$$\begin{aligned} I_0 &= \{+2, +3, \ldots, +N\} \\ I_1 &= F \cup T = \{*, /, \sin, \cos, \exp, r\log, x, R\} \end{aligned} \tag{5}$$

where F = {*, /, sin, cos, exp, rlog} and T = {x,R} are function and terminal set. +N, *, /, sin, cos, exp, rlog, x, and R denote the addition, multiplication, protected division, sine, cosine, exponent, protected logarithm, system inputs, and random constant number, respectively, and take N, 2, 2, 1, 1, 1, 1, 0 and 0 arguments [24]. N is an integer number (the maximum number of an ODE terms), I_0 is the instruction set and the root node, and the instructions of other nodes are selected from the instruction set I_1 [20].

It is worth mentioning that if the right-hand side of ODEs is the polynomial, the instruction set I_1 can be defined as I_1 = {*2, *3, ..., *n, x_1, x_2, ..., x_n, R} [24].

C. Optimization of Model Structure

A hybrid evolutionary method is used to obtain the structure optimization of models, in which GP and PSO are employed to evolve the architecture and the parameters of FNT and the additive tree models, respectively. The proposed method interleaves both optimizations.

Finding an optimal or near-optimal tree model is formulated as a product of evolution. A number of tree variation operators are developed as follows [20, 21]:

(1) Mutation. Three mutation operators were chosen to generate offsprings from the parents. The related mutation operators are described as:

a) Change one terminal node: select a random terminal node in the tree and replace it with another terminal node, which is randomly generated.

b) Grow: select a random leaf in hidden layer of the tree and replace it with a new generated subtree.

c) Prone: randomly select a function node in a tree and replace it with a terminal node selected in the set T.

(2) Crossover. First two parents are selected based on the predefined crossover probability P_c and select one nonterminal node in the hidden layer for each tree randomly, and then swap the selected subtree.

(3)Selection. EP-style tournament selection [25] is applied to select the parents for the next generation. Pairwise comparison is conducted for the union of μ parents and μ offsprings. For each individual, q opponents are chosen uniformly at random from all the parents and offspring. For each comparison, if the individual's fitness is no smaller than the opponent's, it receives a selection. Select μ individuals out of parents and offsprings, which have most wins to form the next generation. This is repeated in each generation until a predefined number of generations or the optimal structure is found.

D. Optimization of Model Parameters using PSO

According to the number of parameters of each tree model, the particles are randomly generated initially. Each particle x_i represents a potential solution. A swarm of particles moves through space with the moving velocity of each particle represented by a velocity vector v_i [20]. At each step, each particle is evaluated and keeps tracking its best position, which is associated with the best fitness that has achieved so far in a vector $Pbest_i$ [21]. The best position among all the particles is kept as *Gbest*. The new velocity for particle *i* is updated by

$$v_i(t+1) = v_i(t) + c_1 r_1 (Pbest_i - x_i(t)) + c_2 r_2 (Gbest(t) - x_i(t)) \tag{6}$$

where c_1 and c_2 are positive constant, and r_1 and r_2 are uniformly distributed random number in [0,1]. Based on the updated velocities, each particle changes its position according to the following equation:

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{7}$$

E. Fitness Function

To obtain an optimal FNT and additive tree model, the following fitness function, the normalized mean squared errors, is given by

$$NMSE = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x_i})}{\frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x})} \tag{8}$$

where x_i, $\overline{x_i}$ and $\hat{x}$ denote the actual, output of FNT model or additive tree model, and average traffic data respectively.

F. The General Learning Algorithm

The general learning procedure for designing a FNT or ODE model is described as follows:

1) Creating the initial population (flexible neural trees or addictive tree model, and their corresponding parameters);

2) Structure optimization by neural tree variation operators as described in subsection *C*, in which the fitness is calculated by *NMSE*.

3) If the better structure is found, then go to step 4), otherwise go to step 2);

4) Parameter optimization is achieved by PSO described in subsection *D*. In this stage, the tree structure is fixed, and the optimal tree is obtained from the end of run of the similar to GP search. All the parameters used in the optimal tree formulate a parameter vector to be optimized by PSO;

5) If the maximum number of iterations of PSO algorithm is reached, or no better parameter vector is found for a significantly long time (100 steps) then go to step 6); otherwise go to step 4);

6) If satisfactory solution is found, then stop; otherwise go to step 2.

G. Ensemble Method

In general, combining the outputs of several predicting models could improve the performance of the single one, which is based on a suitable decomposition of the prediction error [26-31]. Expected ensemble members must be accurate and diverse, which poses the problem of generating a set of predictors with reasonably individual performances and independently distributed predictions for the test points [26].

FNT and ODEs are viewed as the feasible and efficient predictors with completely different components. To improve the accuracy of time series forecasting, the following methods are introduced to integrate the outputs of optimal FNT and ODEs model (*N* is the number of the predictors, $f_k(x)$ is the output of the ensemble model).

(1) The basic ensemble method

The basic ensemble method (BEM) output is defined as:

$$f_{BEM} = \frac{1}{N}\sum_{k=1}^{N} f_k(x) \tag{9}$$

Although this approach leads to improve the performance, it does not take into account the fact that some models may be more accurate than others. It has the advantage that it is easy to understand and implement.

(2) The generalized ensemble method

A generalization to the BEM method is to find weights for each output that minimize the positive and negative classification rates of the ensemble [26]. The general ensemble method (GEM) is defined as:

$$f_{GEM} = \frac{1}{N}\sum_{k=1}^{N} \alpha_k f_k(x) \tag{10}$$

where $\alpha_k(x)$ is chosen to minimize the normalized mean squared errors between the predictor outputs and the desired values. The optimal weights of the ensemble predictors are optimized by using PSO algorithm.

TABLE I.
PARAMETERS OF EXPERIMENT

Parameters	Value
Population size	50
Crossover probability	0.06
Mutation probability	0.05
PSO population size	50
PSO maximum iterations	100
PSO c_1	2.0
PSO c_2	2.0
PSO v_{max}	5.0

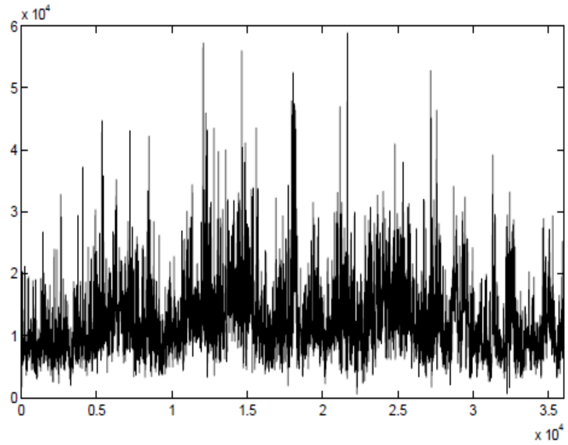

Figure 4. Filtered traffic measurements data

III. EXPERIMENTAL RESULTS AND ANALYSIS

In general, the traffic measurements are considered as a sum of a regular process and a stochastic part which are related to the high-frequency noise [32, 33]. Because the elimination of the noise may simplify the analyzed time series, the wavelet soft threshold noise reduction method is applied to deal with those data [21]. The difference between the original time series and the filtered signal are caused by the noisy component. The original traffic series, the corresponding filtered signal, and the noisy components are presented in Figs. 1, 4, and 5, respectively. Then, the filtered traffic measurements data are normalized in the interval [0, 1] with the following formula

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{11}$$

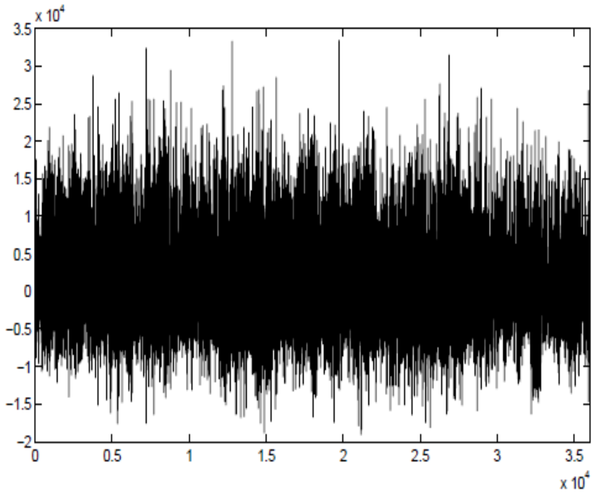

Figure 5. Noisy component

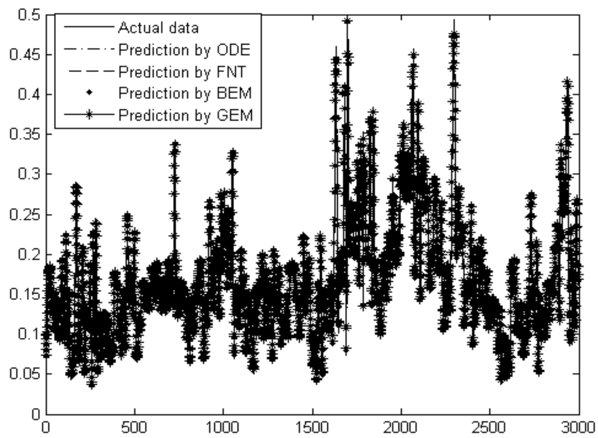

Figure 6. Comparison of actual data and the predicted ones by four methods.

The length of this traffic data set is 36000. The front 33000 data points are used as the training set and the last 3000 data points are used as the test set. Table I shows the parameters of experiment. In the paper, we use the front 10 variable to predict the current variable. The used instruction set for additive tree model $I_0 = \{+_2, +_3, +_4, +_5, +_6, +_7, +_8\}$ and $I_1 = \{*, /, log, exp, rlog, sin, cos, X_1, X_2, X_3, \ldots, X_9\}$. The instruction sets used to create an optimal FNT model is $S = F \cup T = \{+_2, +_3, \ldots, +_6\} \cup \{x_1, \ldots, x_9\}$. And the experiments are carried out on a single-CPU personal computer (Pentium III 933 MHz).

To access the performance of the different ensemble paradigms by quantifying the prediction obtained on an independent data set, the root mean squared error (RMSE), maximum absolute percentage error (MAP) and mean absolute percentage error (MAPE) are used to study the performance of the trained forecasting model for the test data, defined as:

$$\begin{aligned} RMSE &= \sqrt{\frac{1}{N}\sum_{k=1}^{N}(y_{actual,k} - y_{predict,k})^2} \\ MAP &= \max\left(\frac{|y_{actual,k} - y_{predict,k}|}{y_{predict,k}} \times 100\right) \\ MAPE &= \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(\frac{|y_{actual,k} - y_{predict,k}|}{y_{predict,k}} \times 100\right)} \end{aligned} \tag{12}$$

where N is the total number of the data, $y_{actual,k}$ is the *k- th* actual data, and $y_{predic,k}$ is the *k-th* forecast value.

Table II and Table III summarize the results achieved using five different methods, and a comparison of actual network traffic time series data and the predicted ones by four different methods is shown in Fig. 6. From Table II and Table III, it can be clearly seen that ensemble methods can effectively improve the accuracy of small-time scale network traffic prediction. The generalized ensemble method performs better than the basic ensemble one.

The prediction error of GEM is shown in Fig. 7. Observing from Fig. 7, GEM could effectively predict the traffic data by using FNT and ODE models.

The statistical distributions of the traffic measurements and the prediction values by GEM are shown in Fig. 8 and Fig. 9. From Figs. 8 and 9, we can see that the time series generated by the ensemble method have very similar probability distribution with the actual traffic measurements time series. Therefore, two ensemble methods can reproduce the statistical distribution of the real traffic measurements data. From Fig. 10, which depicts the statistical distributions of the absolute difference between the actual time series and the predicted data by GEM, it can be clearly seen that the prediction error of GEM for the traffic data mainly concentrates on the vicinity of zero.

TABLE II.
NMSE OF FOUR LEARNING METHODS

NMSE	Training data	Testing data
Neural network	0.092304	0.073236
FNT	0.062945	0.011215
ODE	0.062014	0.011147
BEM	0.059123	0.009157
GEM	0.048845	0.007324

TABLE III.
STATISTICAL ANALYSIS OF FOUR LEARNING METHODS (TEST DATA)

Criterion	FNT	ODE	BEM	GEM
MAP	159.46	157.3	155.82	140.3
MAPE	5.01	4.84	4.51	4.00
RMSE	0.0129	0.0124	0.0119	0.0101

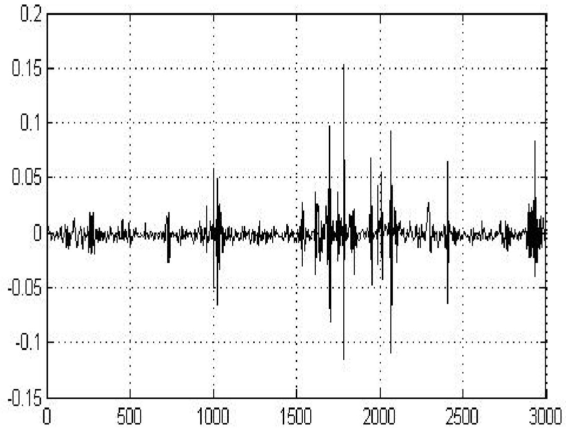

Figure 7. The predicted errors

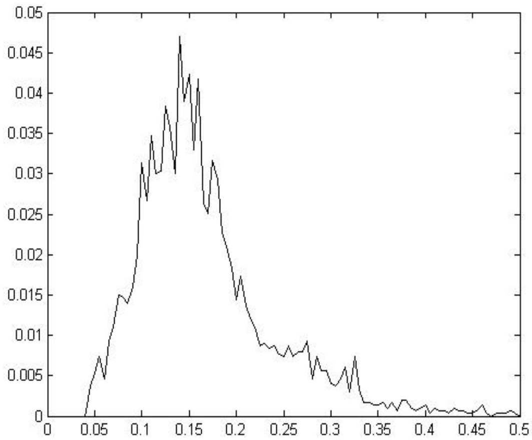

Figure 8. The statistical distributions of the traffic measurements data.

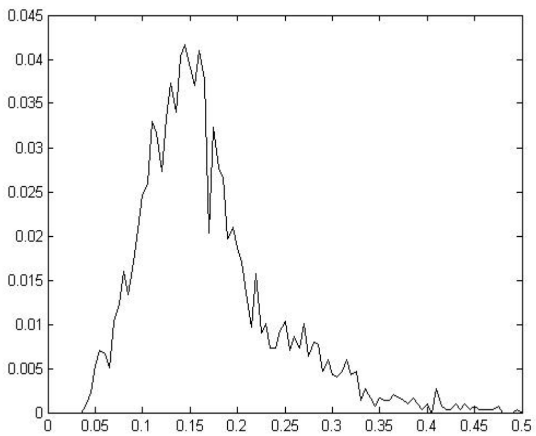

Figure 9. The statistical distributions of the prediction data.

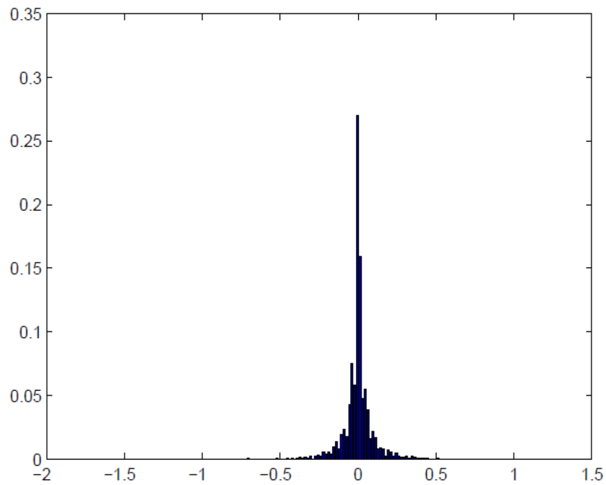

Figure 10. The statistical distributions of the predicted errors.

IV. CONCLUSIONS

In this paper, the ensemble of flexible neural tree model and ordinary differential equations has been proposed to predict the traffic measurement data. By using the flexible neural tree model and ordinary differential equations as wholly different and accurate predictors, the ensemble members are accurate and diverse. The architecture and the parameters of FNT and the additive tree models are evolved by GP and PSO, respectively. The experimental results show that the ensemble method can be successfully used for the prediction of traffic measurements data compared with other three methods. The prediction error converges to the small value, even at zero, and the prediction accuracy of the generalized ensemble method is superior to the basic ensemble one.

ACKNOWLEDGMENT

This research is partially supported by the Natural Science Foundation of China (60573065), the Natural Science Foundation of Shandong Province (No.ZR2010FM040), the Natural Science Foundation of Province (Y2007G33), the Shandong Distinguished Middle-aged and Young Scientist Encourage and Reward Foundation of China (BS2009SW003), and the Key Subject Research Foundation of Shandong Province.

REFERENCES

[1] I. Gasser, G. Sirito, B. Werner, “Bifurcation analysis of a class of car-following traffic models”, *Physica D*, vol. 197, pp.222-241, 2004.

[2] G. Orosz, B. Krauskopf, R.E. Wilson, “Bifurcations and multiple traffic jams in a car-following model with reaction-time delay”, *Physica D*, vol. 211, no. 3-4, pp.277-293, 2005.

[3] G. Orosz, R.E. Wilson, B. Krauskopf, *Bifurcations in a car-following model with delay in Fifth IFAC Workshop on Time-Delay Systems*, W. Michiels and D. Roose eds, IFAC, 2004.

[4] A.D. Doulamis, N.D. Doulamis, S.D. Kollias, “An adaptable neural-network model for recursive nonlinear traffic prediction and modeling of MPEG video sources”,

Neural Networks, IEEE Transactions on, pp. 150-166, 2003.
[5] Y.B. Xie , W.X. Wang , B.H. Wang , "Modeling the coevolution of topology and traffic on weighted technological networks", *Phys. Rev. E*, vol. 75, 026111, 2007.
[6] S. Uglig, "Non-stationarity and high-order scaling in TCP flow arrivals:a methodological analysis", *ACM SIGCOMM computer communications*, vol. 34, no. 2, pp.9-24, 2004.
[7] Z.L. Zhang, V.J. Ribeiro, S. Moon, C. Diot, "Small-time scaling behaviors of Internet backbone traffic: an empirical study", *IEEE Computer and Communications Societies*, vol. 3, pp.1826-1836, 2003.
[8] H. Xiao, H. Sun, B. Ran, Y. Oh, "the Fuzzy-neural network traffic prediction framework with wavelet decomposition, *Transportation Research Record*, pp.16-20, 2003.
[9] G.Q. Mao, "Real-Time Network Traffic Prediction Based on a Multiscale Decomposition", *Lecture Notes in Computer Science*, vol. 3420, pp.492-499, 2005.
[10] D.S.K. Karunasinghe, S.Y. Liong, "Chaotic time series prediction with a global model: artificial neural network", *J. Hydrol*. Vol. 323, pp.92-105, 2006.
[11] A. Freking, W. Kinzel, I. Kanter, "Learning and predicting time series by neural networks", *Phys. Rev. E*, vol. 65, pp.050903, 2005.
[12] M. Small, C.K. Tse, "Minimum description length neural networks for time series prediction", *Phys. Rev. E*, vol. 66, pp.066701, 2002.
[13] Q.L. Ma, Q.L. Zheng, H. Peng, T.-W. Zhong, J.-W. Qin, "Multi-step-prediction of chaotic time series based on co-evolutionary recurrent neural network", Chin. *Phys. B*, vol. 17, no. 2, pp.536-542, 2008.
[14] H. Liu, D. Liu, L.F. Deng, "Chaotic time series prediction using fuzzy sigmoid kernel-based support vector machines", *Chin. Phys*, vol. 15, no. 6, pp. 1196-1200, 2006.
[15] H. Li, J. Zhang, X. Xiao, "Neural Volterra filter for chaotic time series prediction", *Chin. Phys*., vol. 14, no. 11, pp.2181-2188, 2005.
[16] Q. Meng, Q. Zhang, W. Mu, "A novel multi-step adaptive prediction method for chaotic time series", *Acta Phys. Sin*., vol. 55, no. 4, 1666-1671, 2006.
[17] B. Zhou, D. He, Z. Sun, "Traffic Modeling and Prediction using ARIMA/GARCH Model", *Modeling and Simulation Tools for Emerging Telecommunication Networks,* pp. 101-121, 2006.
[18] Q.F. Meng, Y.H. Chen, Y.H. Peng, "Small-time scale network traffic prediction based on a local support vector machine regression model", *Chinese Physics B*, vol. 18, no. 6, pp.2196, 2008.
[19] P. Wu, Y.H. Chen, Q.F. Meng, Z. Liu, "Small-Time Scale Network Traffic Prediction Using Complex Network Models", *2009 Fifth International Conference on Natural Computation*, Tianjin, 2009, pp.303-307,.
[20] Y.H. Chen, B. Yang, Q.F. Meng, Y.O. Zhao, A. Abraham, "Time-series forecasting using a system of ordinary differential equations", *Information Sciences*, vol. 181, no. 1, pp.106-114, 2010.
[21] Y.H. Chen, B. Yang, Q.F. Meng, "Small-time Scale Network Traffic Prediction Based on Flexible Neural Tree", *Applied Soft Computing*, vol. 12, pp.274-279, 2012.
[22] Traffic archive, http://ita.ee.lbl.gov/.
[23] A. Abraham, N.S. Philip, P. Saratchandran, "Modeling chaotic behavior of stock indices using intelligent paradigms", *Int. J. Neural Parallel Sci. Comput*., vol. 11 no. 1-2, pp.143-160, 2003.
[24] Y.H. Chen, J. Yang, Y. Zhang, J.W. Dong, "Evolving Additive tree models for System Identification", *International Journal of Computational Cognition*, vol. 3, no. 2, pp.19-26, 2005.
[25] K. Chellapilla, "Evolving computer programs without subtree crossover", *IEEE Transactions on Evolutionary Computation*, vol. 1, pp.209-216, 1997.
[26] Y.H. Chen, B. Yang, A. Abraham, "Flexible Neural Trees Ensemble for Stock Index Modeling", *Neurocomputing*, vol. 70, no. 4-6, pp.697-703, 2007.
[27] A.J.C. Sharkey (Ed.), Combining Artificial Neural Nets, Springer, London, 1999.
[28] S. Geman, E. Bienenstock, R. Doursat, "Neural networks and the bias/variance dilemma", *Neural Comput*, vol. 4, pp.1-58, 1992.
[29] G. Zhang, J. Yin, X.M. He, L,L. Cheng, "Data Dependant Learners Ensemble Pruning", *Journal of Software*, vol. 7, no. 4, pp. 919-926, 2012.
[30] C. Ribeiro, R. Goldschmidt, R. Choren, "A Reuse-based Environment to Build Ensembles for Time Series Forecasting", *Journal of Software*, vol. 7, no. 11, pp. 2649-2656, 2012.
[31] M. Li, J.L Gao, "An Ensemble Data Mining and FLANN Combining Short-term Load Forecasting System for Abnormal Days", *Journal of Software*, vol. 6, no. 6, pp. 961-968, 2011.
[32] P. Shang, X. Li, X, S. Kamae, "Chaotic analysis of traffic time series", *Chaos Solitons Fractals*, vol. 25 pp. 121-128, 2005.
[33] P. Shang, X. Li, S. Kamae, "Nonlinear analysis of traffic time series at different temporal scales", *Phys. Lett. A*, vol. 357, pp. 314-318, 2006.

Bin Yang is currently pursuing his Ph.D. in School of Information science and Engineering from Shandong University, Jinan, P.R. China. He received his B.Sc. and Master degree in School of Information Science and Engieering from University of Jinan. His research interests include Hybrid Computational Intelligence and its applications in time-series prediction, system identification and Gene regulatory network**.**

Mingyan Jiang is a professor in School of Information Science and Engineering of Shandong University, Jinan, P.R. China. He received his M. S. degree in 1992 in electrical engineering and got his Ph. D degree in Communication Engineering from Shandong University in 2005 in China. His research interests include communication signal processing, Soft computing methods, Image processing, communication network, remote control and navigation tracking and positioning technology.

Yuehui Chen was born in 1964 in Shandong Province of China. He received his B.Sc. degree in mathematics/automatics from the Shandong University of China in 1985, and Master and Ph.D. degree in electrical engineering from the Kumamoto

University of Japan in 1999 and 2001. During 2001-2003, he had worked as the Senior Researcher of the Memory-Tech Corporation at Tokyo. Since 2003 he has been a member at the Faculty of Electrical Engineering in Jinan University, where he is currently head of the Laboratory of Computational Intelligence. His research interests include Evolutionary Computation, Neural Networks,Fuzzy Logic Systems, Hybrid Computational Intelligence and their applications in time-series prediction, system identification, intelligent control, intrusion detection systems, web intelligence and bioinformatics. He is the author and co-author of more than 70 technique papers.Professor Yuehui Chen is a member of IEEE, the IEEE Systems, Man and Cybernetics Society and the Computational Intelligence Society, a member of Young Researchers Committee of the World Federation on Soft Computing, and a member of CCF Young Computer Science and Engineering Forum of China. More information at: http://cilab.ujn.edu.cn.

Qing-fang Meng, IEEE Member, was born in 1979. She received her M. S. degree in 1992 in signal and information processing and got her Ph. D degree in Communication Engineering from Shandong University in China. Since September 2008, she has engaged in teaching and research in the School of Information Science and Engineering from University of Jinan. Her research interests include computational intelligence, biological information processing, nonlinear dynamics research.

Ajith Abraham received the M.S. degree from Nanyang Technological University, Singapore, and the Ph.D. degree in Computer Science from Monash University, Melbourne, Australia. He is currently the Director of Machine Intelligence Research Labs (MIR Labs), which has members from more than 80 countries. He works in a multidisciplinary environment involving machine intelligence, network security, various aspects of networks, e-commerce, Web intelligence, Web services, computational grids, data mining, and their applications to various real-world problems. He has given more than 50 plenary lectures and conference tutorials in these areas.