Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/092L41

A Small Scale Forecasting Algorithm for Network Traffic based on Relevant Local Least Squares Support Vector Machine Regression Model

Tao Peng and Zhoujin Tang[*]

School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, 100876 Beijing, China

Received: 9 Aug. 2014, Revised: 10 Nov. 2014, Accepted: 11 Nov. 2014
Published online: 1 Apr. 2015

Abstract: Real-time monitoring and forecasting technology for network traffic continues to play an important role in network management. Effective network traffic prediction detects and avoids potential overload problems before they occur, which significantly improves network availability and stability. Recent research has centered around Time Series Analysis based traffic prediction methods that primarily extend Neural Network Forecasting (NNF) and Least Squares Support Vector Machine (LSSVM) algorithms, which are not without their drawbacks. Given the vulnerabilities of existing nonlinear prediction methods in forecasting modeling, this paper presents a novel, Relevant Local (RL) forecast method and its accompanying Pattern Search (PS) parameter-optimization approach to introduce a new small-scale network traffic forecasting algorithm called RL-LSSVM (Relevant Local-Least squares support vector machine). Furthermore, we demonstrate our new algorithm on network traffic data collected from wired campus networks and show that the RL-LSSVM can effectively predict the small scale traffic measurement data while exhibit significantly improved prediction accuracy than existing algorithms.

Keywords: Small scale network traffic, Time series forecasting, Least squares support vector machine, Pattern search

1 Introduction

Computer networking is one of the fastest growing areas in computer science, not in small part due to the fact that the services and applications carried over networks grow ever richer and more sophisticated. Improvements in network management enable administrators and users to better leverage existing networks through improved, efficient utilization of network speed and utilization. This goal is one of the key drivers of intelligent network traffic prediction research and technology: accurate prediction of the network traffic is useful in understanding of network dynamics, so it has great significance for network designs and flow control.

Generally, forecasting time horizons determine which long-term and short-term forecasting algorithms are used; for example, traffic with longer time horizons use long-term forecasting algorithms, and the vice versa in the case of shorter time horizons. Existing research has primarily focused on long and medium time horizon forecasting [1,2,3,4,5,6,7,8,9,10,11,12,13], where the sampling interval of traffic flow are on the order of days [1,2,3], hours [4,5,6] and minutes [7,8,9,10,11,12,13].

One classical problem of linear time series prediction methods over long time horizons is that of uncertainty; the longer the time horizon, the greater the degree of uncertainty. For a network that carries conventional data, this uncertainty is mitigated through the use of adequately large training datasets. However, with the services and applications carried by the network growing increasingly sophisticated, changes in network traffic have become more and more complex, often compromising the accuracy of existing linear prediction methods.

The basis of non-linear prediction methods is as follows: if the network traffic characteristics on the time scale and the prediction model could be analyzed, controlling the network traffic during development trend of time can be achieved and network congestion would be handled appropriately. Non-linear methods needed to achieve three objectives: First, since the network environment users change their behavior complexity and variability, the network environment changes rapidly.

[*] Corresponding author e-mail: tangzhoujin@gmail.com

Thus the forecasting algorithm must be able to adapt to network usage in an automated manner. Second, network traffic analysis and forecasting should be capable of filtering and ignoring abnormal data to reduce the impact of noise on the overall prediction. Third, communication networks tend to possess patterns of self-similarity and long-range correlations and the forecasting algorithm should be capable of leveraging these patterns to accurately forecasting nonlinear changes and compatibility of noise.

Various of nonlinear time series prediction algorithm have been proposed [5,10,11,12,13,14,15,16], but they each has its unique drawbacks. For example, the body of neural network based methods [13,14,16], are known to pose at least three challenges: first, the structure of neural networks are difficult to design; second, neural networks tend to exhibit from slower convergence rate compared to alternatives; third, neural networks pose the risk of over-fitting input data.

Given their success in other application spaces with similar characteristics, Support Vector Machine (SVM) models were introduced to incorporate Structural Risk Minimization Principle (SRMP) and Statistical Learning Theory (SLT) into the prediction process with the goal of overcoming the challenges posed by traditional neural network based methods [17]. Of particular note is the Least Squares Support Vector Machine (LSSVM) proposed by Sunkens and Vandewall [18]. The LSSVM is notable because it simplifies the inequality constraint to an equality constraint and can be trained to solve a set of linear equations. This simplification lies at the heart of why LSSVM has been widely adopted in the field.

Despite the attractive features of the above calculations, LSSVM forecasting algorithms still have some drawbacks: first, LSSVM forecasting algorithms consider all the data to be useful, making LSSVM vulnerable to outlier effects of abnormal data; second, LSSVM model parameters have a great impact on predictive performance but they are notoriously difficult to optimize.

In this paper, we propose a novel short-term network traffic flow forecasting algorithm to solve those problems that improves the LSSVM method in two major ways. First, our algorithm leverages the self-similarity of network traffic data series to select the training set which is best correlated with forecast data, thereby avoiding the problem of computing high dimension inverse matrix by reducing the length of training data set. Second, our algorithm automatically selects optimized combination of parameters accurately by pattern search algorithm which is simple and effective.

2 Relevant local LSSVM prediction model

2.1 Relevant local forecast method

Nonlinear time series forecast methods can be divided into two categories, Global Forecast Method (GFM) and Local Forecast Method (LFM) [19]. GFM uses all available data in its prediction and fits the system by dynamic equations to the entire dataset. As such, neural network forecast methods and traditional SVM forecast methods are Global Forecast Method because both families of methods train forecast models using all input-output pairs. Generally, the computation and complexity of GFM is large. In contrast, LFM uses only select subsets of historic data to forecast future values. Consequently, LFM computation and complexity are lower than that of GFM. The disadvantage of LFM is that aside from data local to a timepoint of interest, all residual historical data are discarded, which means that traditional LFM methods do not fully utilized the self-similarity of input historical data series to maximize prediction accuracy.

Consequently, LFM computation and complexity are lower than that of GFM. The disadvantage of LFM is that aside from data local to a timepoint of interest, all residual historical data are discarded, which means that traditional LFM methods do not fully utilized the self-similarity of input historical data series to maximize prediction accuracy.

The Relevant local forecast model training set selection operators are as follows:

(1) Input origin historical time series $X = x_1, x_2, \cdots, x_n; x \in R$.

(2) Segment historical time series by local minima, which are endpoints of time series segmentation to define location sequence M.

(3) Select a data segmentation$P$, which is composed of points near x_n; namely, $\left[x_{n-s+1}, x_{n-s+2}, \cdots, x_n\right]$. The length of data segmentation s is determined by analysis of the time series, specifically X's autocorrelation coefficients C. We define the distance from each start point to first minimum point of the coefficients series C as the length of data segmentation s.

(4) Calculate the Euclidean distance series data segmentation D of P from origin history time series X using the sliding window method. This yields and expression of D as follows:

$$D(i) = \sqrt{\sum_{j=1}^{s} \left(P(j) - X(i+j-1)\right)^2}, \quad (i = 1, \cdots, n-s) \tag{1}$$

(5) Sort Euclidean distance series data based on ascending order to obtain the sorted index sequence I.

(6) Find the max value of $M \mid M \le I(1)$ and the minimum value of $M \mid M \ge I(1) + s - 1$. Select a data segmentation which makes the max value of $M \mid M \le I(1)$

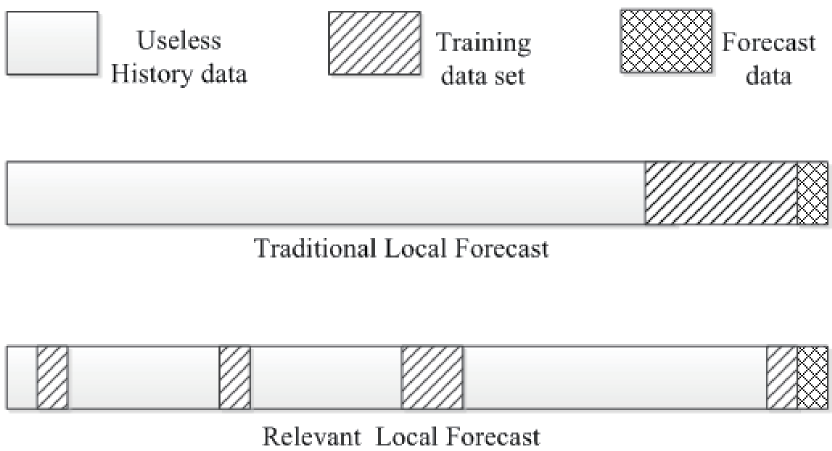

Fig. 1: The difference of Local Forecast and Relevant Local Forecast

the start location and make the minimum value of $M \mid M \geq I(1)+s-1$ the end location, and chooses it as training sequence Xtrain. If the length of training sequence X_{train} is smaller than the threshold (default: 200), add 1 to index of I, algorithm continues to select a data segmentation, which make the max value of $M \mid M \leq I(2)$ the start location and make the minimum value of $M \mid M \geq I(2)+s-1$ the end location, else algorithm uses the union of training sequence and the selected data segmentation as new training sequence. Repeat the above step until the length of training sequence is greater than or equal to threshold.

2.2 Least Squares Support vector Machine Regression

Given a one-dimensional training data series that can be represented as a training load matrix S as:

$$\begin{bmatrix} X_{train}(1) & X_{train}(2) & \cdots & X_{train}(m) \\ X_{train}(2) & X_{train}(3) & \cdots & X_{train}(m+1) \\ \vdots & \vdots & & \vdots \\ X_{train}(l-m+1) & X_{train}(l-m+2) & \cdots & X_{train}(l) \end{bmatrix}$$

So training data set is input data, $y_k \in R$ is output data and l is the number of total data points. For the nonlinear relationship between x'_k and y_k, LSSVM maps x'_k to a higher dimensional space R^h, and regresses training data in the high-dimensional space:

$$f(x') = w^T\varphi(x') + b, (w \in R^h, b \in R) \tag{2}$$

Where $\varphi(x')$ is nonlinear mapping function, w is normal vector, and b is offset amount. Equation 2 can be expressed as:

$$\min_{w,b,e} J(w,e) = \frac{1}{2}w^T w + \frac{\lambda}{2}\sum_{k=1}^{n} e_k^2 \tag{3}$$

Equation 3 subject to:

$$y_k = w^T\varphi(x'_k) + b + e_k, (k = 1, \cdots, n) \tag{4}$$

Where $e_k \in R, (k = 1, \cdots, n)$, are error variables, and $\lambda \geq 0$ is a regularization constant. To solve the optimization problem, we can introduce the Lagrangian function:

$$\begin{aligned} L(w,b,e,\alpha) = \frac{1}{2}w^T w + \frac{\lambda}{2}\sum_{k=1}^{n} e_k^2 - \\ \sum_{k=1}^{n}\alpha_k(w\varphi(x'_k) + b + \xi_k - y_k) \end{aligned} \tag{5}$$

Where $\alpha_k, (k = 1, 2, \cdots, n)$ are Lagrange multipliers.

From Karush-Kuhn-Tucker (KKT) conditions we can derive the following relationship:

$$\begin{bmatrix} 0 & \mathbf{1}_{n\times1}^T \\ \mathbf{1}_{n\times1} & \Omega + \frac{I}{\lambda} \end{bmatrix} \times \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ Y \end{bmatrix} \tag{6}$$

Where $\alpha = [\alpha_1, \alpha_2, \cdots, \alpha_n]^T$, $\mathbf{1}_{n\times1}^T$ is a $n \times 1$ vector where all the elements are equal to 1, Ω is an $n \times n$ symmetrical matrix, and $\Omega_{ij} = k(x'_i, x'_j) = \varphi(x'_i)^T\varphi(x'_j)$, $y = [y_1, y_2, \cdots, y_n]^T$, $k(\cdots)$ is the kernel function.

Some typical kernel functions are:

Linear:$k(x, x_k) = x^T x_k$

Polynomial:$k(x, x_k) = (x^T x_k + 1)^d$

Radial-basis function:$k(x, x_k) = exp(-\|x - x_k\| \setminus \sigma^2)$

Defining $Q_n = \Omega + \frac{I}{\lambda}$, and from (7) we obtain the solution of b and α:

$$\begin{aligned} b &= \frac{\mathbf{1}_{n\times1}^T Q_n^{-1} y}{\mathbf{1}_{n\times1}^T Q_n^{-1}\mathbf{1}} \\ \alpha &= Q_n^{-1}(y - \mathbf{1}_{n\times1} \times b) \end{aligned} \tag{7}$$

Substituting b and α into (1), we obtain the result LSSVM model for function estimation follows as:

$$y(x') = \sum_{k=1}^{n}\alpha_k k(x', x'_k) + b \tag{8}$$

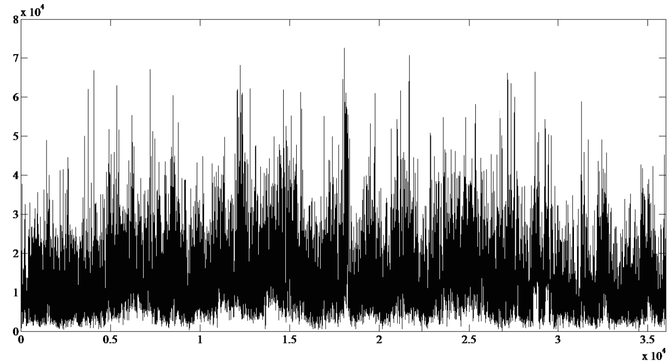

Fig. 2: Traffic data aggregated within 0.1 s

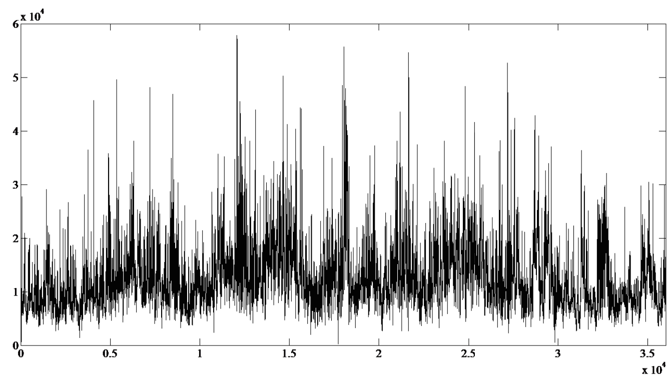

Fig. 3: Filtered traffic data

2.3 Parameter optimization by Pattern search

Pattern search method is efficient in low-dimensional optimization problem, making it a good choice for LSSVM parameter optimization [20]. We choose the standard pattern search algorithm to select LSSVM parameters and the specific implementation steps can be found in the references [20].

For the parameters selection of LSSVM-based forecasting model, we define the objective function f as follows:

(1) Define the validation sample $X_v = [x_{n-m+1}, x_{n-m+2}, \cdots, x_n]$ by the leave-one-out method and the length of X_v is m.

(2) Set initial $i = 1$ and make the remaining data $[x_1, x_2, \cdots, x_{n-m+(i-1)}]$ the candidate set of training set.

(3) Contain the training set X_{train} form the candidate set by the Relevant Local forecast method proposed in this paper.

(4) Input X_{train} and parameters $\mathbf{q}$ into LSSVM forecasting model and get the forecasting value $\widehat{x_{n-m+i}}$.

(5) If $i \leq m, i = i+1$, and repeat (2) through (4); else, go to (6).

(6) The value of the objective function $f(\mathbf{q})$ is $\sqrt{\frac{\sum_{i=1}^{m}(X_v(i)-\widehat{X_v(i)})^2}{m-1}}$, in which $\widehat{X_v(i)}$ is the forecast value of $X_v(i)$ by the Relevant local LSSVM forecast method.

3 Experimental result and analysis

This section demonstrates the performance of the proposed algorithm in the case of real network traffic. The real network traffic used in the experiment is from the Lawrence Berkeley National Laboratory. The dataset consists of an hour's worth of all wide-area traffic between Digital Equipment Corporation and the rest of the world. Detailed information about the traffic trace can be found in their webpage (http://ita.ee.lbl.gov/). The data package used in this paper is DEC-Pkt1[21], and the time stamps have millisecond precision.

First, we aggregate traffic data with time bin 0.1 s, that is, the arrived data bytes are distributed within the 0.1 s time interval.

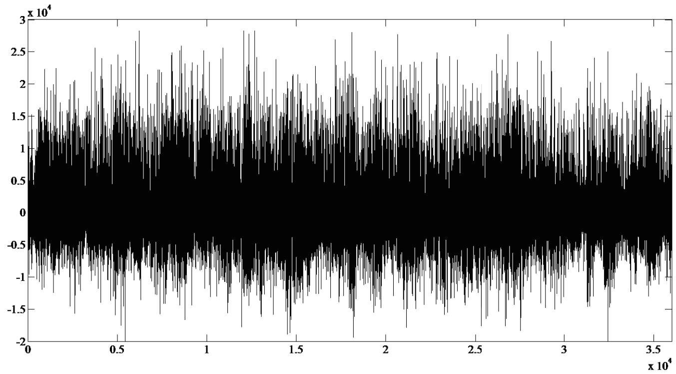

Fig. 4: Noisy component

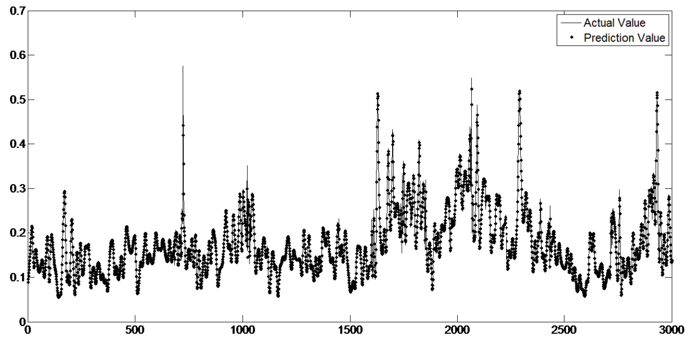

Fig. 5: Comparison of actual network traffic time series data and the predicted ones

Second, we normalized the data with following formula.

$$x_{normalized} = (x - x_{min})/(x_{max} - x_{min}) \tag{9}$$

The traffic data are shown as Fig.2.

The length of this traffic historical data sequence is 36,000. The first 33,000 data points are used as the training set, and the last 3000 data points are used as the test set. In order to facilitate comparison of performance, those sets same as data sets which are used in references 14 through 16.The normalized mean squared error is used to evaluate the prediction accuracy as follows:

$$NMSE = \frac{(1/L)\sum_{i=1}^{L}|x_{N+i} - \widehat{x_{N+i}}|^2}{(1/N)\sum_{i=1}^{N}|x_i - \overline{x}|^2} \tag{10}$$

We select data series $[x_{32000}, x_{32001}, x_{32002}, \cdots, x_{33000}]$ as validation sample by the leave-one-out method, and use pattern search method which is based on LSSVM prediction and relevant local training method to search the optimal combination of parameters. The optimized parameters for the prediction of validation sample were as follows: embedding dimension $m = 7$, delay $\tau = 1$, regularization constant of LSSVM model $\lambda = 4122$, variance of RBF kernel function $\delta^2 = 37$.

We input the optimized parameters into our RL-LSSVM prediction model and our prediction results are shown in Fig.5 with prediction errors in Fig.6. The distributions of the RPE (Relative Percent Error), [(prediction-observation)/ observation*100], for the one step are shown in Fig. 7.

In Fig.5, the solid line represents the original data and the asterisk represents predicted data traffic, which fit the actual traffic value very well. This demonstrates that our proposed algorithm is a high accurate and anomaly-tolerant strategy. Moreover, the NMSE of prediction is as small as 3.4459E-3. Fig.6 and Fig.7 presents the error value and the distributions of the Relative Percent Error (RPE), those can be clearly seen that the RPE of more than 95 points are less than 5

From the tabulated, it is indicated that our proposed RL-LSSVM traffic forecasting algorithms outperforms other forecasting approaches in [14,15,16] with promising results. The results obtained indicate that, better forecasting accuracy is possible when the training

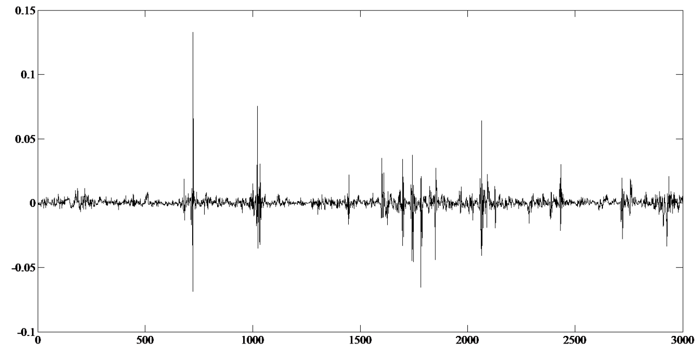

Fig. 6: The prediction error

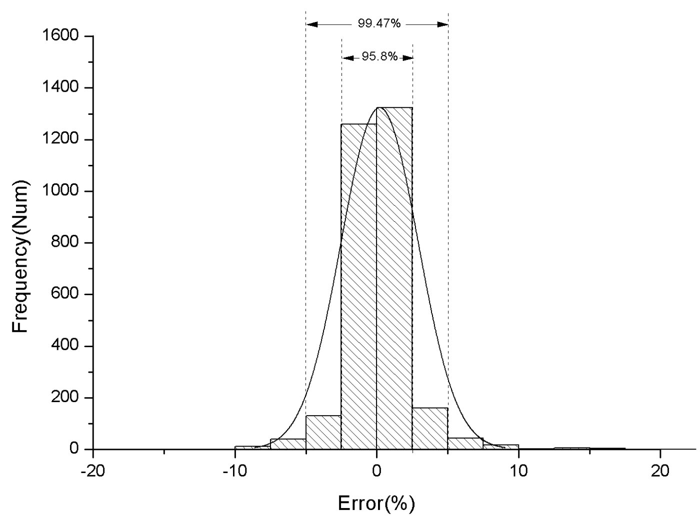

Fig. 7: The distributions of the prediction RPE

Table 1: Comparison of proposed algorithm with recently proposed method

No.	Method	NMSE	Reference
1	Flexible Neural Tree+PSO	$1.1215*10^{-2}$	14
2	Feed-forward Neural Network+PSO	$7.32*10^{-2}$	14
3	Local Relevance Vector Machine+PSO	$8.5*10^{-3}$	16
4	Local Linear Prediction	$1.43*10^{-2}$	16
5	Local SVM	$1.0135*10^{-2}$	15
6	Relevant Local LSSVM+PS	$3.4459*10^{-3}$	

data has a strong autocorrelation which is contained from the selection by the Relevant Local Training method.

Furthermore, in method 1-3, the prediction models select parameters by the Particle Swarm Optimization algorithm, but the fitness function of PSO is the error of test data set. In other words, those parameters selections are a posteriori optimization methods and cant be achieved before forecasting in the real case. Method 4-5 pre-determined parameters of the prediction models. We also use PS method to search optimal parameters of validation sample which contain from historical data set and use those parameters to predict test data. Finally, our method is easy to implement in the real case.

4 Conclusion

In this paper a nonlinear prediction algorithm based on a relevant local LSSVM regression model is proposed to predict small scale network traffic and Pattern Search is applied to forecasting model parameters optimization LSSVM forecasting models structure is fixed and easy to regulate. The relevant local prediction method selects training set from all the historical data to ensure that self-similarity of the training set and filters out irrelevant data of test set. Using a validation sample and PS method makes the algorithm automatically select forecasting model parameters before predicting test data. Our Relevant Local LSSVM technique is applied on the prediction of short-term network traffic data series. We demonstrate the great effectiveness and efficiency of both prediction technique and parameter optimization in contrast to existing methods, and show that the prediction error mainly concentrates on the vicinity of zero.

Acknowledgement

This work is supported by the Project supported by the National Science and Technology Major Project of the Ministry of Science and Technology of China (Grant No.2012ZX03003011).
The authors are grateful to the anonymous referee for a careful checking of the details and for helpful comments that improved this paper.

References

[1] M. C. Tan ,S. C. Wong, J. M. Xu, Z. R. Guan, and P. Zhang, IEEE Transaction on Intelligent Transportation Systems **10**, 60-69 (2009).
[2] M. Lippi, M. Bertini and P. Frasconi, IEEE Transaction on Intelligent Transportation Systems **14**, 871-881 (2013).
[3] A. Maria, R. Matias, A. Macedo, P.R.M. Maciel, and L.B. Araujo, 12th International Conference on Parallel and Distributed Computing, Applications and Technologies, Korea, 109-115 (2011).
[4] J. Jiang and S. Papavassiliou, Computer Communications **29**, 1627-1638 (2006).
[5] R. Li, J. Y. Chen, Y. J. Liu and Z. K. Wang, The Journal of China Universities of Posts and Telecommunications **17**, 88-93 (2010).
[6] C. N. Manoel, Y. S. Jeong, M. K. Jeong and D. H. Lee, Expert Systems with Applications **36**, 6164-6173 (2009).
[7] E. Vlahogianni, M. Karlaftis and J. Golias, Transportation Research Part C **13**, 211-234 (2005).
[8] H. Chang, Y. Lee, B. Yoon and S. Baek, IEEE Transaction on Intelligent Transportation Systems **6**, 292-305 (2011).
[9] T. T. Tigran, B. Biswajit and O. M. Margaret, IET Intelligent Transport Systems **13**, 519-526 (2012).
[10] R. C. Bao and F. T. Hsiu, Expert Systems with Applications **36**, 6960-6965 (2009).
[11] H. L. Sun, Y. H. Jin, Y.D. Cui and S. D. Cheng, Journal of Network and Systems Management **18**, 427-447 (2011).
[12] X. W. Liu, X. Fang, Z. Qin, C. Ye and M. Xie, Chinese Physics B **19**, 4760-4768 (2009).
[13] R. C. Bao and F. T. Hsiu, Applied Soft Computing **9**, 1177-1183 (2009).
[14] Y. H. Chen, B. Yang and Q. F. Meng, Applied Soft Computing **12**, 274-279 (2012).
[15] Q. F. Meng Q F, Y. H. Chen, Z. Q. Feng, F. L. Wang and S. S. Chen, Chinese Physics B **18**, 2194-2199 (2009).
[16] Q. F. Meng Q F, Y. H. Chen and Z. Q. Feng, Acta Physica Sinica **62**, 1-7 (2013).
[17] V. N. Vapnik, The Nature of Statistical Learning Theory. 2nd ed, USA: Springer, (1999).
[18] N. I. Sapankevych and R. Sankar, IEEE Computational Intelligence Magazine **4**, 24-38 (2009).
[19] J. D. Farmer and J. J. Sidorowich, Physical Review Letters **59**, 845-848 (1987).
[20] L. G. Zhou, K. K. Lai and L. Yu, Soft Computing **13**, 149-155 (2009).
[21] D. Peter, M. Jeff, P. Vern and S. Mike, Internet traffic archive, 2013-09-10, http://ita.ee.lbl.gov/.

Tang Zhoujin is currently a Ph.D. candidate at Beijing University of Posts and Telecommunications (BUPT), Beijing, China. His research interests include spectrum sensing, cognitive radio networks, radio resource allocation for wireless networks and signal processing.

Peng Tao received the B.S., M.S. and Ph.D. degrees from School of Telecommunications Engineering, Beijing University of Posts and Telecommunications (BUPT), Beijing, China. After that, he joined the Nokia Research Center (Beijing) as a research engineer, where he was engaged in the research on planning and optimizing for WCDMA and UTRA-TDD HSDPA networks. His research interests include 3G and long term evolution (LTE) system, wide-band wireless access networks, cognitive radio networks, software de fined radio (SDR), embedded systems, and network planning/optimizing.