Vol 18 No 11, November 2009
1674-1056/2009/18(11)/4760-09

Chinese Physics B

© 2009 Chin. Phys. Soc.
and IOP Publishing Ltd

Network traffic prediction by a wavelet-based combined model*

Sun Han-Lin(孙韩林)[a)†], Jin Yue-Hui(金跃辉)[a)],
Cui Yi-Dong(崔毅东)[a)b)], and Cheng Shi-Duan(程时端)[a)]

[a)] *State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing* 100876, *China*

[b)] *School of Information and Telecommunication Engineering, Beijing University of Posts and Telecommunications, Beijing* 100876, *China*

(Received 23 February 2009; revised manuscript received 20 March 2009)

Network traffic prediction models can be grouped into two types, single models and combined ones. Combined models integrate several single models and thus can improve prediction accuracy. Based on wavelet transform, grey theory, and chaos theory, this paper proposes a novel combined model, wavelet–grey–chaos (WGC), for network traffic prediction. In the WGC model, we develop a time series decomposition method without the boundary problem by modifying the standard à trous algorithm, decompose the network traffic into two parts, the residual part and the burst part to alleviate the accumulated error problem, and employ the grey model GM(1,1) and chaos model to predict the residual part and the burst part respectively. Simulation results on real network traffic show that the WGC model does improve prediction accuracy.

Keywords: network traffic prediction, wavelet transform, grey model, chaos model
PACC: 0650, 0270

1. Introduction

Network traffic prediction plays an important role in network management, planning, traffic congestion control, traffic engineering, etc. There are long-term predictions and short-term ones. We study short-term prediction in this paper.

Generally, time series prediction methods can be grouped into two types, single models[1−8] and combined ones.[9−26] Combined models integrate several single models to improve prediction accuracy. Most combined models are decomposition based, namely, they first decompose the original time series into isolated components, secondly, they employ a single model to predict each of them, and lastly they integrate the results to gain the final prediction value.[9−23] Because of the approximate one-fold property of each component, their prediction will be improved. Some others use more than one single model to predict the original series independently, then compute the weighted averages of the results.[24,25] Reference [26] uses chaos theory to optimize the neural network parameters for network traffic forecasting.

It is well known that network traffic shows multiple properties, such as trend, period, self-similar, and long-range dependence, etc. Studies have also revealed that network traffic is chaotic.[27] A combined model can capture many more of these properties than a single model, and thus might improve prediction accuracy. In this paper, we propose a novel wavelet-based combined model, wavelet–grey–chaos (WGC), to predict network traffic. We develop a signal decomposition method based on the à trous algorithm to avoid the boundary problem of wavelet transform. Additionally, in order to alleviate the problem of accumulated error, in WGC we decompose the traffic series into two parts, the residual part and the burst part, and use the grey model and chaos model to predict them respectively.

The remainder of this paper is organized as follows: Section 2 surveys the related work of wavelet-based network traffic prediction; Section 3 introduces wavelet transform, grey theory, and chaos theory, and outlines the WGC model. Section 4 validates the WGC model on real network traffic data. Finally, Section 5 presents the conclusion.

*Project supported by National Basic Research Program of China (Grant Nos 2009CB320505 and 2009CB320504) and National High Technology Research and Development Program of China (Grant Nos 2006AA01Z235, 2007AA01Z206 and 2009AA01Z210).
†Corresponding author. E-mail: sunhanlin@gmail.com

2. Related works

In combined prediction models, the most commonly used decomposition method is wavelet transform for its simplicity. There are continuous wavelet transforms and discrete wavelet transforms. Since in practice time series are usually discrete, the discrete wavelet transform is widely used. The two popular discrete wavelet transform algorithms are the Mallat algorithm and the à trous algorithm. References [12]–[18] adopt the former and Refs.[19]–[23] use the latter.

The major difference between these two algorithms is that the Mallat algorithm is shift-variable, while the à trous algorithm is shift-invariable. Shift-variable means that if we delete (or add) a few values from (or to) the front or the end of the input series, the decomposition results would be not the same as heretofore. The reason is that the Mallat algorithm takes down the sampling action after each decomposition, but the à trous one does not. For practical prediction, the shift-invariable algorithm is preferred since the predicted value and the values used to establish the prediction model should stay in the same series.

Another problem associated with wavelet transform is the so-called boundary problem, namely, that the decomposed values near both edges are distorted.[28,29] Because the input series is finite, when computing the decomposed coefficients near an edge, some artificial values should be interpolated. As a result, the related coefficients are not the same as the real ones. The most commonly used extension methods include zero padding, constant padding, symmetry padding, linear interpolation, etc. In Ref.[30], Feng and Shu argued that constant extension is better than others for network traffic. In Ref.[20], Peng *et al* used predicted values as the extension to alleviate the boundary problem. However, no matter what values are used as the surrogates, there is discrepancy between the interpolated values and the real ones. For prediction applications, the last few values are of critical importance as most prediction models, such as the auto regressive model, auto regressive moving average model, and neural network, use these values as input. The prediction could be harmed if distorted values are used.

In Ref.[23], Ding and Zhou improved the à trous algorithm to avoid the boundary problem by trimming the distorted values at both edges and shifting the remainder right to align with the previous level. However, this may change the shape of the remained signal dramatically, or lead to large detail coefficients, especially for periodic series. For example, the valley (or the peak) value could be occasionally shifted to being the surrogate of the peak (or the valley) one.

In fact, if the Haar function is used in wavelet transform, the boundary problem can be avoided.[13,14,19] The problem is that the residual signal is not smooth (stepwise). Based on the à trous algorithm, we develop a signal decomposition method without the boundary problem while keeping the residual smooth.

The number of decomposition levels is another problem in wavelet-based models. In general, the more decomposition levels there are, the more accurate each component prediction is. However, too many decomposed components could cause a large accumulated error, i.e., the sum of each component prediction error. In the WGC model, we first decompose the network traffic into several levels, then add up all the details to be the burst part. In other words, we decompose the traffic into two parts, the burst part and the residual part, to avoid the accumulated error problem.

3. WGC model

The WGC model is based on wavelet transform, grey theory, and chaos theory. We first introduce all the three theories involved, then outline the complete WGC model.

3.1. Wavelet transform

As mentioned earlier, the à trous algorithm is a shift-invariable wavelet transform algorithm. Another advantage of this algorithm is that the length of each decomposed component is same as that of the original series, so it is easy to relate the information of each level to any given time point. The corresponding drawback is that the storage requirement increases. Here we only give a brief introduction (for more details, refer to Ref.[31]).

3.1.1. À trous algorithm overview

In keeping with the literature, we denote the original time series as $\{c_{0,t}\}$ and the discrete low-pass filter as h. At the level $j+1$, the smoothed coefficient $c_{j+1,t}$ at time t is

$$c_{j+1,t} = \sum_k h(k) c_{j,t-2^j k}, \tag{1}$$

and the corresponding wavelet coefficient is

$$w_{j+1,t} = c_{j,t} - c_{j+1,t}. \tag{2}$$

From Eq.(1), it can be seen that after each decomposition at level j, the low-pass filter h should be dilated by upsampling, i.e., inserting zero after each coefficient of the filter. As a result, the length of h increases by a factor of 2.

Through iteration, the original series could be decomposed into a set of wavelet coefficients plus the residual ones (the smoothed data of the last level). Supposing the maximum decomposition level is J, for any time t, the signal could be expressed as

$$c_{0,t} = c_{J,t} + \sum_{j=1}^{J} w_{j,t}. \tag{3}$$

Equations (1) and (2) provide the decomposition formulas, and Eq.(3) gives the reconstruction one.

3.1.2. À trous-based decomposition method

Following the Haar function wavelet transform, we slightly modify the standard à trous algorithm to avoid the boundary problem.

Suppose we compute the coefficient $c_{j+1,t}$. In the standard algorithm, after reversing the filter, the middle of h is aligned with the coefficient $c_{j,t}$ of the previous level. Instead, we shift the reversed filter and make its right edge align with $c_{j,t}$. This means we only use values less than t in calculating $c_{j+1,t}$, thus the boundary problem is removed. The cost is that the decomposed results may lose the perfect properties of wavelet transform and may not be used for signal analysis.

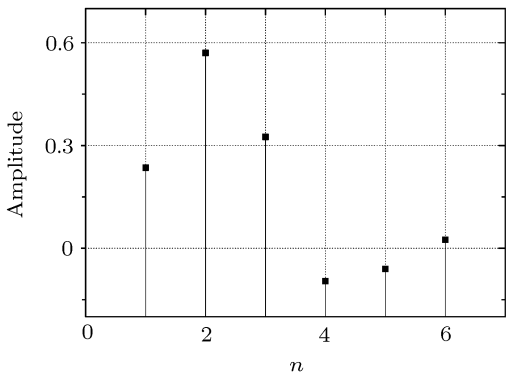


Fig.1. Daubechies filter, order 3.

For the filter used, we select that of the Daubechies wavelet function because it is asymmetric as shown in Fig.1. The asymmetric property makes it possible to give the coefficients near $c_{j,t}$ heavy weight, thus may avoid large wavelet coefficients. Figure 2 plots the decomposed results of a network traffic series. The decomposition level and the order of the Daubechies filter are both 3.

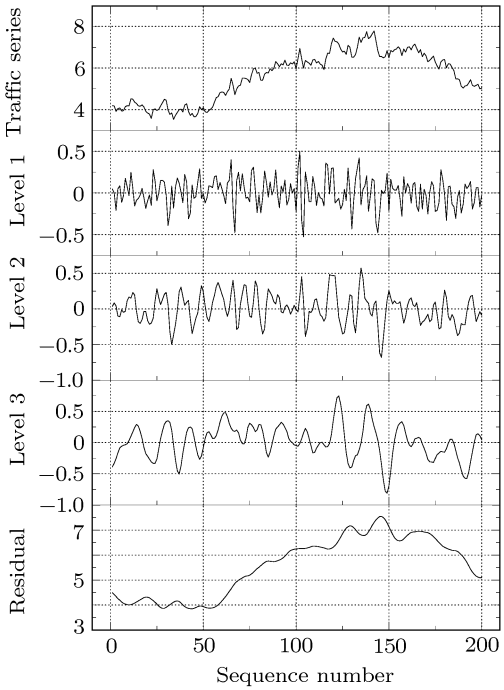


Fig.2. Decomposition illustration.

3.2. Grey model

Grey theory is superior in investigating the inherent law of systems with imprecise and incomplete information. Its major advantages are simple calculation and precise prediction with small amount of data.[32,33]

The grey model is represented by GM(n, h), where n is the rank of different equations in the model and h is the number of variables. GM(1,1) is the simplest one and is widely used for prediction.

Again, for consistency with the literature, let $\boldsymbol{X}^{(0)} = (x^{(0)}(1), x^{(0)}(2), \ldots, x^{(0)}(n))$ be the original time series, and $\boldsymbol{X}^{(1)} = (x^{(1)}(1), x^{(1)}(2), \ldots, x^{(1)}(n))$ be its 1 accumulated generating operation (1-AGO) resulting series, where $x^{(1)}(k) = \sum_{i=1}^{k} x^{(0)}(i)$. Denote the 1 inverse accumulated generating operation (1-IAGO) of the original series as $\alpha^{(1)}\boldsymbol{X}^{(0)} = (\alpha^{(1)}x^{(0)}(2), \alpha^{(1)}x^{(0)}(3), \ldots, \alpha^{(1)}x^{(0)}(n))$, where $\alpha^{(1)}x^{(0)}(k) = x^{(0)}(k) - x^{(0)}(k-1)$, and the adjacent averaging series as $\boldsymbol{Z}^{(0)} = (z^{(0)}(2), z^{(0)}(3), \ldots, z^{(0)}(n))$, where $z^{(0)}(k) = (z^{(0)}(k-1) + z^{(0)}(k))/2$. Then the

GM(1,1) is as follows:

$$x^{(0)}(k) + a \times z^{(0)}(k) = b, \tag{4}$$

and its whitenization function is

$$\frac{\mathrm{d}x^{(1)}(t)}{\mathrm{d}t} + a \times x^{(1)} = b. \tag{5}$$

Denoting the coefficient a and the parameter b together as a vector $\hat{\boldsymbol{\alpha}} = [a, b]^{\mathrm{T}}$ means that they can then be solved by the least-mean-square (LMS) algorithm, $\hat{a} = (\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{Y}$, where

$$\boldsymbol{Y} = \begin{pmatrix} x^{(0)}(2) \\ x^{(0)}(3) \\ \cdots \\ x^{(0)}(n) \end{pmatrix}, \quad \text{and} \quad \boldsymbol{B} = \begin{pmatrix} -z^{(0)}(2) & 1 \\ -z^{(0)}(3) & 1 \\ \vdots & \vdots \\ -z^{(0)}(n) & 1 \end{pmatrix}.$$

By solving Eq.(5), we obtain

$$\hat{x}^{(1)}(k+1) = \left(x^{(0)}(1) - \frac{b}{a}\right)\mathrm{e}^{-ak} + \frac{b}{a}. \tag{6}$$

Then by 1-IAGO, the modelling value of the original series is

$$\hat{x}^{(0)}(k+1) = \left(x^{(0)}(1) - \frac{b}{a}\right)(1 - \mathrm{e}^{a})\mathrm{e}^{-ak}. \tag{7}$$

We utilize the so-called metabolic GM(1,1) in prediction, namely we only use the latest n data to establish GM(1,1). Suppose the modelling data is $(x^{(0)}(1), x^{(0)}(2), \ldots, x^{(0)}(n))$. To obtain a new measurement $x^{(0)}(n+1)$, we add it at the end of the modelling data, remove $x^{(0)}(1)$, then reestablish the model.

3.3. Chaos theory

Chaos theory assumes that all variables of a nonlinear system are interacting during their evaluations, and each one's dynamic evaluation behaviour is determined by that of others. So it is possible to investigate the system behaviour by studying one of these variables. According to Taken's theorem, the phase space of the original system could be reconstructed through one scalar time series of the system. The most commonly used phase space reconstruction method is delay coordinates.

3.3.1. Phase space reconstruction

For more details about Taken's theorem, refer to Refs.[34]–[36]. Here, we just briefly describe the delay coordinates phase space reconstruction method.

Supposing the measured one-variable scalar time series of an unknown chaotic system is $\boldsymbol{X} = \{x_1, x_2, \ldots, x_N\}$, the reconstructed system phase space could be denoted by

$$\begin{pmatrix} \boldsymbol{Y}(1) \\ \boldsymbol{Y}(2) \\ \vdots \\ \boldsymbol{Y}(M) \end{pmatrix} = \begin{pmatrix} x_1 & x_{1+\tau} & \cdots & x_{1+(m-1)\tau} \\ x_2 & x_{2+\tau} & \cdots & x_{2+(m-1)\tau} \\ \cdots & \cdots & \cdots & \cdots \\ x_M & x_{M+\tau} & \cdots & x_{M+(m-1)\tau} \end{pmatrix},$$

where m is the embedding dimension, τ is the delay time, and $M = N - (m-1)\tau$. The reconstructed phase space is diffeomorphism with the original system. Taken's theorem indicates that the embedding dimension should satisfy $m \geq 2d + 1$, where d is the dimension of the system. We refer to $\boldsymbol{Y}(k)$ as delay vector.

The selection of embedding dimension m and delay time τ is essential because they have a crucial influence on successful analysis. The best τ should not be too large or too small. If τ is too small, the coordinate values of the delay vector have a strong correlation with each other, and then the reconstructed phase space vectors will compress around the main diagonal. In contrast, if τ is too large, the coordinate values of the delay vector will be almost random, and the inherent relationship between the variables of the original system must be destroyed. Similarly, the embedding dimension should also not be too small or too large. If m is too small, the condition in Takens' theorem will not be fulfilled and the phase space cannot be unfolded thoroughly, while if it is too large, practical problems will occur due to the finite amount of data points.[36]

Many methods have been proposed to compute the embedding dimension and the delay time from a scalar time series.[35,37] Generally, they can be grouped into two types. One of them determines the two parameters individually, such as the self-correlation method, multiple autocorrelation method, mutual information method, and so on. The other assumes that the two parameters be correlated and determines them simultaneously, such as the C–C method, time window method, etc. In this paper, we use the method presented in Ref.[37], which computes m and τ concurrently.

3.3.2. Chaos prediction

Based on the reconstructed phase space, a number of prediction methods have been developed for chaotic

time series, such as global-region methods, local-region methods, methods based on the largest Lyapunov exponent, neural networks, etc.[6−8,35] Since the local-region method is stable and robust, it is used widely. For the sake of simplicity, we employ the simplest one, the zero-rank local-region method, in this paper.

The root of the zero-rank local-region method is that the dynamic behaviour of a phase point (delay vector) is affected mostly by its nearest neighbours, hence its next step evaluation could be approximated by its neighbours' next positions. Let the last point of the reconstructed phase space, $\boldsymbol{Y}(M)$, be the centre point. Find $\boldsymbol{K}$ numbers of its nearest neighbouring phase points, $\{\boldsymbol{Y}(i_1), \boldsymbol{Y}(i_2), \ldots, \boldsymbol{Y}(i_K)\}$, and the next position of centre point, $Y(M+1)$, can be predicted by

$$Y(M+1) = \frac{1}{K}\sum_{k=1}^{K} Y(i_k+1). \tag{8}$$

The next value of the original time series, x_{N+1}, which is located at the end of $\boldsymbol{Y}(\boldsymbol{M}+1)$, thus can be separated.

3.4. WGC model

Based on the wavelet transform, the grey model GM(1,1) and the chaos prediction method introduced previously, we describe the complete WGC model here.

Let the network traffic series be $\boldsymbol{X} = \{x_1, x_2, \ldots, x_n\}$. First, decompose $\boldsymbol{X}$ into J levels. Denote wavelet coefficients at level j as $d_{j,t}$ and the smoothed coefficients of the last level as $c_{J,t}$. We add up the wavelet coefficients of all levels to obtain the burst part,

$$b_t = \sum_{j=1}^{J} d_{j,t}. \tag{9}$$

Now we have decomposed the original series into two parts, the burst part $\{b_t\}$ and the residual part $\{c_{J,t}\}$, simply as $\{c_t\}$. Then we utilize the chaos model to predict the burst part because it behaves with chaotic characteristics. For the residual part, we employ the GM(1,1) model to forecast. Though the residual part is also chaotic, the prediction result of GM(1,1) is better than that of the chaos model. In addition, GM(1,1) is much simpler than the chaos model.

Since GM(1,1) can be established by a small amount of data, only a few points need be decomposed. However, establishing the chaos prediction model needs much more historical data. As a result, we have to decompose as many traffic series as the chaos model needs.

Note that the decomposition algorithm used here is shift invariant and free of the boundary condition, so to acquire new measured data, we do not have to decompose all the data needed to build GM(1,1) and chaos models. In fact, if we store the decomposed result of previous steps, just one convolution with a proper filter is needed to compute coefficient $c_{j,t}$.

By now, we have $\hat{b}_{n+1}$ and $\hat{c}_{n+1}$, one step ahead of the predicted values of the burst part and the residual part respectively. By the reconstruction formula of the $\hat{a}$ trous algorithm, the predicted value of the next step traffic is

$$\hat{x}_{n+1} = \hat{c}_{n+1} + \hat{b}_{n+1}. \tag{10}$$

Figure 3 shows the procedure of WGC model prediction.

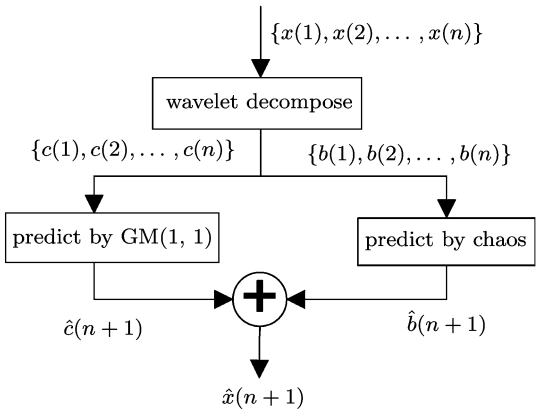


Fig.3. WGC diagram.

4. Validation of the WGC model

In this section, we validate our WGC model by simulation on the real network traffic of Abilene, the internet 2 backbone of America.

4.1. Performance metrics

Several metrics have been used to quantitatively evaluate the prediction performance, including mean relative error, mean square error, normalized mean square error, and signal-error-ratio.[5,24,31] In this paper, we use all four metrics to assess the performance of the WGC model. They are defined as follows:

(i) Mean relative error (MRE)

$$E_{\mathrm{MRE}} = \frac{1}{M}\sum_{t=1}^{M} \frac{|\hat{x}(t) - x(t)|}{x(t)}, \tag{11}$$

where $\hat{x}(t)$ and $x(t)$ are the predicted value and the actual value at time t respectively. M is the total prediction number. As prediction accuracy increases, E_{MRE} decreases.

(ii) Mean square error (MSE)

$$E_{\mathrm{MSE}} = \frac{1}{M}\sum_{t=1}^{M}(\hat{x}(t) - x(t))^2. \tag{12}$$

The smaller the E_{MSE}, the better the prediction.

(iii) Normalized mean square error (NMSE)

$$E_{\mathrm{NMSE}} = \frac{1}{\sigma^2}\frac{1}{M}\sum_{t=1}^{M}(\hat{x}(t) - x(t))^2, \tag{13}$$

where σ^2 is the variance of time series over prediction time duration.

(iv) Signal-error-ratio (SER)

$$E_{\mathrm{SER}} = 10\mathrm{lg}\left[\frac{E(x(t)^2)}{E((x(t) - \hat{x}(t))^2)}\right]. \tag{14}$$

$E(\cdot)$ is the expected value. Apparently, the larger the E_{SER}, the better the prediction accuracy.

4.2. Traffic data

We evaluate the WGC model by the real network traffic of Abilene. The traffic matrixes were collected between Mar. 1, 2004 and Sept. 10, 2004 by Yin Zhang, assistant professor at Texas University. The data can be downloaded from his homepage[1)].

We distribute the traffic matrix to each link according to the routing table, then normalize the traffic of each link to its utilization ratio. We select one set of traffic (link 6, from the 20737th to the 22752th data points) to validate the WGC model. The interval between two successive data points is 5 minutes, thus the duration of the traffic set lasts for one week. The traffic set is shown in Fig.4.

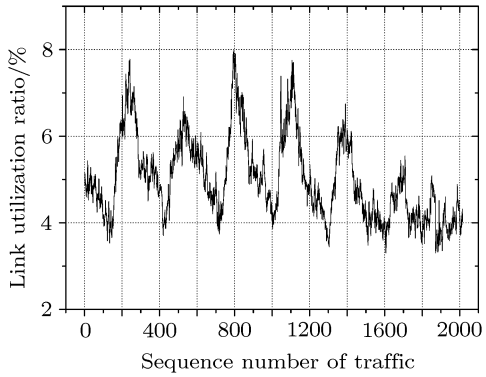


Fig.4. Network traffic set for validation.

[1)] http://www.cs.utexas.edu/yzhang/

4.3. Prediction results

In this section, we discuss the prediction of the WGC model, and compare the results with those of some other models. We predict the last 416 data points of the selected traffic set.

4.3.1. Decomposition

The first step of WGC prediction is to decompose the traffic series. We decompose the original traffic series into the burst part and the residual part. The aim is to separate the traffic's multi-property to some extent. For the selected traffic data, we utilize the developed method to decompose the traffic into 3 levels, then compute the burst part by adding up all the three detailed series. The Daubechies filter used here is of order 3. Figure 5 shows the decomposition results. It can be seen that the residual part retains the periodic property of the traffic series, but it is much smoother than the original one as shown in Fig.4.

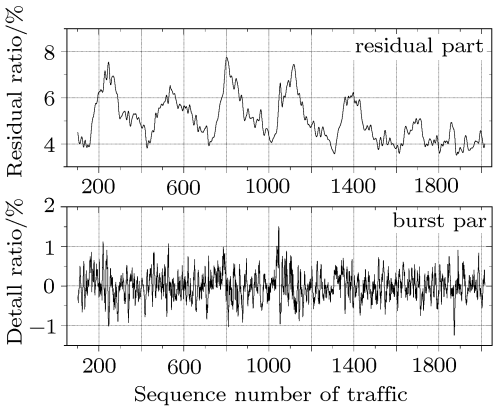


Fig.5. Network traffic decomposition.

Remember that the decomposition method used is shift-invariable and free of the boundary problem, thus it is possible to decompose the overall traffic and store the results before simulation so as to make the simulation simple.

4.3.2. Residual part prediction

We employ GM(1,1) to predict the residual part. In our previous study, we found that the GM(1,1) model is appropriate for predicting time series of which the dominant period length (the number of samples in one dominant period) is much greater than the modelling length (the number of samples of modelling

data). The dominant period refers to the first prominent period. For a traffic series of 5-minute intervals, the dominant period length is 288, the number of samples in one day. We build GM(1,1) of length 4 to predict the residual part. The E_{MRE} of the prediction is merely 0.4466%. In other words, the final accuracy is almost totally dependent on the prediction of the burst part.

4.3.3. Burst part prediction

For the burst part, we use the chaos model for the prediction. Before applying the chaos model, we should first determine whether the system is chaotic or not. Many methods have been proposed to achieve this.[35] The most commonly used one is the largest Lyapunov exponent method: if the exponent is greater than 0, the system is chaotic, otherwise it is not. There are numerous methods to calculate the largest Lyapunov exponent of a scale time series. We use the small amount data method proposed in Ref.[38]. The largest Lyapunov exponent of the burst part is 0.4195, greater than 0, thus the chaos model can be used for prediction.

By now, we can gain the final prediction by using Eq.(10). The performance metrics of the WGC model are listed in Table 1. Some other models' performance metrics are also shown. We will explain them later.

4.3.4. Discussion

In fact, the traffic series itself is chaotic and can be predicted by the chaos model too. However, from Table 1, it can be seen that direct prediction by the chaos model is worse than that by the WGC model. This can be explained by the multi-property of traffic series: via decomposition, these properties are separated to some extent and appropriate models should be employed to predict each of them individually, thus the prediction is improved.

Table 1. Prediction results.

metric	GM(1,1)	chaos	ARIMA	wavelet ARIMA	WGC	WGC*
E_{MRE}	4.3345	4.5475	9.7760	9.0486	3.9511	4.5927
E_{MSE}	0.0547	0.0606	0.2819	0.2388	0.0458	0.0618
E_{NMSE}	0.2789	0.3093	1.4379	1.2183	0.2336	0.3150
E_{SER}	25.1265	24.6758	18.0044	18.7243	25.8961	24.5981

In addition, the decomposition of traffic series has another advantage. By comparing the delay time τ and embedding dimension m of the original traffic series with that of the corresponding burst part, we find that after decomposition, m increases slightly but τ decreases dramatically. We recalculate τ and m each time when modelling data set is updated, i.e. new data is added and the oldest data is deleted. The means of m and τ of the original series are 5.9351 and 260.7139 respectively, while the corresponding means of its burst part are 9.1803 and 9.1803 respectively. Recall that the number of reconstructed phase points is $N-(m-1)\times\tau$, where N is the length of scalar series. Given N and m, small τ will increase the phase point number, and that is a benefit for chaos analysis. In our experiment, though m is increased slightly, τ is reduced strikingly. As a result, the number of phase points is increased.

We believe that the increase of m and the reduction of τ have some relationship with the throw of period by decomposition. After decomposition, the burst part is more chaotic than the original traffic series: the largest Lyapunov exponents of the burst part and the original series are 0.4195 and 0.0207 respectively. According to chaos theory, the embedding dimension should increase. For the reduction of τ, we explain this phenomenon by the autocorrelation method, a traditional delay time calculating method. In this method, when the autocorrelation function decreases to $1-1/e$ of its original values, the corresponding lag is considered as the optimal reconstructed delay time. For series with a period, a small lag will lead to a large autocorrelation value since a minor shift results in a resembled series, and thus a large autocorrelation value. In short, the delay time τ of a series associates with the length of its period, and

by removing the period, the delay time τ will reduce. Since the period length of a 5-minute interval traffic series is very large, 288, the extent of the τ reduction is dramatic.

4.3.5. Comparison

We compare the prediction results of the WGC model with that of several other models, including GM(1,1), the chaos model, auto regressive integrated moving average (ARIMA), and wavelet-based ARIMA models. The results are listed in Table 1.

We list the prediction of GM(1,1) and the chaos model as they are the two used in the WGC model and we want to show that the prediction is indeed improved. From Table 1, we can see that all the four metrics of WGC model are superior. Here, the modelling length of GM(1,1) is 6, which we find is the best for prediction of 5-minute averaging network traffic. The chaos model is the zero-rank local-region model, the same as used in the WGC model. Its modelling length is 1500.

ARIMA is a classic time series prediction model. However, Table 1 indicates that the ARIMA prediction is worse than that of GM(1,1) and chaos for network traffic series. The modelling length of ARIMA is also 1500, and its parameters are calculated using matlab script doarima[1)].

In the wavelet-based ARIMA model, we decompose the traffic into 3 levels by our method and predict each component by the ARIMA model. Since different characteristics are captured by individual ARIMA models, the prediction accuracy is improved. However, as Table 1 shows, the prediction of wavelet-based ARIMA is still worse than that of the WGC, GM(1,1) and chaos models. Comparing the prediction results of the combined model with that of their associated single models, we can conclude that the combined model can improve prediction to some extent, but the basic level of prediction accuracy is determined by the single models used.

The WGC* model is similar to wavelet-based ARIMA except that we utilize the chaos model instead of ARIMA to predict each detailed component and GM(1,1) to predict the residual part. Unfortunately, the results are even worse than those of chaos model and GM(1,1). This is because the accumulated error exceeds the error of these two single models, and explains why we decompose the traffic into two parts.

5. Conclusion

Network traffic behaves with multi-properties. Combined models integrate several single models and can capture more of these properties than single models, thus can improve the prediction accuracy. In this paper, we propose a combined model, WGC, based on the à trous wavelet transform algorithm, grey theory, and chaos theory, for network traffic prediction.

We slightly modify the standard *à* trous wavelet transform algorithm to avoid the boundary problem. The cost is that the perfect properties of the wavelet transform may be lost. In order to alleviate the accumulated error problem, we decompose the traffic series into two parts, the burst part and the residual part by adding all the details. We employ GM(1,1) and the chaos model to predict the residual part and the burst part respectively. The simulation results for real network traffic show that the WGC model does improve prediction accuracy.

References

[1] Zou B X and Liu Q 2002 *J. Comput. Res. Dev.* **39** 1645 (in Chinese)

[2] Zare Moayed H and Masnadi-Shirazi M A 2008 *International Symposium on Information Technology* Kuala Lumpur, Malaysia, August 26-29, 2008 p1

[3] Shu Y T, Wang L, Zhang L F, Xue F, Jin Z G and Yang O 2001 *Chin. J. Comput.* **24** 46 (in Chinese)

[4] Cao J H, Liu Y and Dai Y 2007 *International Conference on Wireless Communications, Networking and Mobile Computing* Shanghai, China, September 21-23, 2007 p6353

[5] Chen S Y, Zhou Y H and Wang W 2005 *Proceedings of International Conference on Machine Learning and Cybernetics* Guangzhou, China, 18-21 August 2005 p3585

[6] Zhang Y and Guan W 2009 *Acta Phys. Sin.* **58** 756 (in Chinese)

[7] Wang Y S, Sun J, Wang C J and Fan H D 2008 *Acta Phys. Sin.* **57** 6121 (in Chinese)

[8] Meng Q F, Peng Y H, Qu H J and Han M 2008 *Acta Phys. Sin.* **57** 1423 (in Chinese)

[9] Cheng G, Gong J and Ding W 2002 *Acta Electron. Sin.* **30** 1633 (in Chinese)

[10] Yao Q F, Li C F, Ma H L and Zhang S 2007 *J. Zhejiang Univ. (Sci. Ed.)* **34** 396 (in Chinese)

[1)] http://www.stat.unc.edu/faculty/hurd/stat185Data/progdoc.html

[11] Yang Y F, Ren X M, Qin W Y, Wu Y F and Zhi X Z 2008 *Acta Phys. Sin.* **57** 6139 (in Chinese)
[12] Li J, Hou X H and Han Z J 2007 *J. Electron. Inf. Technol.* **29** 725 (in Chinese)
[13] Hong F and Wu Z M 2006 *Chin. J. Comput.* **29** 166 (in Chinese)
[14] Hu J, Hu Y Q and Xiao Z Q 2008 *Comput. Eng.* **34** 112 (in Chinese)
[15] Xu K, Xu J W and Ban X J 2001 *Acta Electron. Sin.* **29** 567 (in Chinese)
[16] Gao G Y, Wu W Y and Deng A Y 2008 *Commun. Technol.* **4** 93 (in Chinese)
[17] Ren Q L 2008 *J. Chongqing Jiaotong Univ. (Nat. Sci.)* **27** 630 (in Chinese)
[18] Jiang C W, Hou Z J, Li T and Zhang Y C 2002 *J. Shanghai Jiaotong Univ.* **36** 1053 (in Chinese)
[19] Renaud O, Starck J L and Murtagh F 2005 *IEEE Trans. Syst. Man Cybernetics* (Part B) **35** 1241
[20] Peng Y, Goldenberg A and Bi Z Q 2001 *KDD 2001 Workshop on Temporal Data Mining in the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, California, USA, August 26-29, 2001
[21] Sinha N, Lai L L, Ghosh P K and Ma Y N 2007 *International Conference on Intelligent Systems Applications to Power Systems, ISAP 2007* Toki Messe, Niigata, November 5-8 2007 p1
[22] Zeng F Z and Qiu Z D 2004 *Signal Process.* **20** 108 (in Chinese)
[23] Ding N and Zhou X Z 2007 *J. Syst. Simul.* **19** 4082 (in Chinese)
[24] Feng H L, Chen D, Lin Q J and Chen C X 2006 *WiCOM 2006. International Conference on Wireless Communications,Networking and Mobile Computing* Wuhan, China, September 22-24, 2006 p1
[25] Ma G S and Bai Y 2008 *Microelectron. Comput.* **25** 153 (in Chinese)
[26] Lu J J and Wang Z Q 2006 *Trans. Nanjing Univ. Aeronautics Astronautics* **23** 316
[27] Liu D L and Shuai D X 2003 *Acta Electron. Sin.* **12** 1866 (in Chinese)
[28] Yuan L H and Song J S 2006 *Appl. Res. Comput.* **3** 25 (in Chinese)
[29] He Y Y, Chu F L, Wang Q Y and Zhong B L 2002 *J. Vib. Eng.* **15** 228 (in Chinese)
[30] Feng H F and Shu Y T 2005 *Proceedings of International Conference on Wireless Communications, Networking and Mobile Computing* Wuhan, China, September 23-26, 2005 p1041
[31] Shensa M J 1992 *IEEE Trans. Signal Process.* **40** 2464
[32] Deng J L 1989 *J. Grey Syst.* **1** 1
[33] Liu S F, Dang Y G and Fang Z G 2004 *Grey System Theory and its Application* (3rd ed) (BeiJing: Science Press) p125-145 (in Chinese)
[34] Takens F 1980 *Phys. Rev. Lett.* **45** 712
[35] Lv J H, Lu J A and Chen S H 2005 *Chaos Time Sequence Analysis and Application* (Wuhan: WuHan University Press) p57 (in Chinese)
[36] Suykens J A K and Vandewalle J P L 1998 *Nonlinear Modeling-Advanced Black-Box Techniques* (Boston: Kluwer Academic Publishers) p209-239
[37] Ma H G, Li X H, Wang G H, Han C Z, Xu J F and Zhu X F 2004 *J. Xi'an Jiaotong Univ.* **38** 335 (in Chinese)
[38] Rosenstein M T, Collins J J and De Luca C J 1993 *Physica* D **65** 117