

Research Article

Two-Phase Model of Multistep Forecasting of Traffic State Reliability

Jufen Yang ¹, Zhigang Liu ², Guiyan Jiang,³ and Lin Zhu¹

¹Dr., College of Urban Railway Transportation, Shanghai University of Engineering Science, China

²Prof., College of Urban Railway Transportation, Shanghai University of Engineering Science, China

³Prof., Faculty of Maritime and Transportation, Ningbo University, China

Correspondence should be addressed to Zhigang Liu; 10060001@sues.edu.cn

Received 7 November 2017; Revised 29 January 2018; Accepted 21 May 2018; Published 9 July 2018

Academic Editor: Aura Reggiani

Copyright © 2018 Jufen Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Multistep prediction of traffic state is a key technology for advanced transportation information system. The research results based on the principle of multistep prediction can provide more information about the traffic operating quality in advance. Considering that prediction error increases with the increasing numbers of multistep predictions, this research proposes the concept of dynamic predictability that is related to the characteristic of historical traffic flow data used. The traffic flow is characterized by randomness, regularity, and volatility according to the traffic flow theory. Therefore, three key indexes are firstly calculated to measure the characteristics of reliability series. Then a two-phase model is established based on wavelet neural network optimized by particle swarm optimization. The upper phase is a model to estimate the number of predictable steps, and the lower phase is the multistep prediction model of reliability. Compared with that of backpropagation neural network and support vector machine, results show that the convergence time of the wavelet neural network optimized by particle swarm optimization is the lowest, which only costs 256 and 291 seconds in both two-phase models under the same conditions. The average relative error of multistep prediction reached the lowest value, 8.91% and 12.01%, respectively, for weekday and weekend data used. Moreover, the prediction performance based on weekday is better than that of weekend. The research results lay a decision-making basis for managers in determining the key parts of road network to develop future improvement measures.

1. Introduction

In recent decades, the increasing demand for road transportation has negatively affected the stability and reliability of traffic operation which caused a series of drawbacks, such as extensive waste of travel time, decreasing of environmental quality, and aggravated vehicle wear and tear. Meanwhile, the requirement for accurate prediction of traffic state is increasing. Prediction results serve not only as an important basis for traffic control and guidance but also as a decision support for travelers to adjust their travel plan.

Development of Advanced Traveler Information Systems usually divides traffic state into three states: smooth, congested, and blocked. It has oversimplified the problem since traffic state parameter is a continuous variable, e.g., flow, density, and speed. This study presents a new concept to describe traffic state. It is the traffic state reliability which

is defined as the degree of actual traffic flow relative to the free flow, which assumes that the reliability of free flow is highest. In order to get the traffic flow trend in the future, the forecasting of traffic state reliability is needed to quantitatively measure the reliability of each alternative in future.

For example, there are five links A, B, C, D, and E for the travelers. The prediction result shows that links A, C, and D are in smooth state. However, the travelers do not know how to determine which is the best link among links A, C, and D. If the reliability prediction could provide the level of actual traffic flow relative to the most reliable free flow state of each link (that is, 0.9, 0.2, 0.92, 0.79, and 0.1). Obviously, the traveler will prioritize link C with the highest reliability value to travel.

Antonioni et al. [1] present an approach for local traffic state estimation and traffic state prediction; the method exploits all available (traffic and other) information and

uses data-driven computational techniques. The approach is advantageous as it can flexibly incorporate additional explanatory variables. Given that previously proposed models can outperform current state-of-the-art models, integrating them into existing traffic estimation and prediction models is valuable. Current technologies, such as global position system- (GPS-) enabled cell phones, can record vehicle trajectories and have opened a new way of collecting traffic data. Hiribarren and Herrera [2] present and assess a new method to estimate traffic states on arterials based on trajectory data. The method is based on the Lighthill–Whitham–Richards theory. Preliminary analysis based on microsimulation suggested that this method yields good traffic state estimates at congested and uncongested situations. Mannini et al. [3] explore traffic state estimation on freeways in urban areas and route-based data to properly feed a second-order traffic flow model. This model is recursively corrected by an extended Kalman filter. Considering the lack of real-time information, these authors use simulation-based data to improve the traffic state estimation accuracy.

In recent years, traffic data are currently collected through various sensors, including loop detectors, probe vehicles, cell phones, Bluetooth devices, video cameras, remote sensing applications, and public transport smart cards. Nantes et al. [4] develop a new model-based methodology for real-time traffic state estimation in urban corridors from multiple sources, particularly from loop detectors and partial observations from Bluetooth and GPS devices. Zheng and Su [5] develop a novel algorithm based on compressed sensing theory to recover traffic data with Gaussian measurement noise, missing partial data, and corrupted noise using information recovered from noisy traffic data and traffic state estimation. These authors extend traffic state estimation method to handle traffic state variables of high dimensions.

Traffic state estimation is a key problem with considerable implication for modern traffic management. Kong et al. [6] propose a novel approach to efficiently estimate and predict urban traffic congestion using floating car trajectory data. An innovative traffic flow prediction method using particle swarm optimization is used to calculate the traffic flow parameters for predicting traffic congestion. Then, a congestion state fuzzy division module is applied to convert the predicted flow parameters to the cognitive congestion state of citizens.

Travelers should be able to select the best path to avoid traffic congestion. However, if travelers also avoid the link with high variability, then travelers will enjoy additional benefits, that is, the “reliability benefits.” To date, traffic reliability has introduced the idea of reliability into traffic research and is an important field of traffic problems analysis. Considerable research has been conducted on traffic reliability, covering from theory to practice and from model to algorithm. Frameworks for reliability analysis have also been developed.

Wang et al. [7] comprehensively review the literature on traffic reliability. Basic definitions, theory, and methods are depicted accompanied with the application of traffic optimization. These authors also discuss the future development of traffic reliability research. Xiao et al. [8, 9] consider the

influence of travel time variability on the congestion profile in the presence of endogenous congestion and the role of scheduling preferences. The results show that the cost of travel time variability is the same in exogenous or endogenous congestion for two classes of preferences: linear at work and constant-exponential marginal utility of time. He et al. [10] introduce road segment and network congestion indexes to, respectively, measure the congestion levels of urban road segment and network. He also carries out a traffic congestion analysis for Beijing expressway network, based on the speed performance data.

Probabilistic forecasting of reliability can be used for risk-averse routing. Bezuglov and Comert [11] have studied the possible applications and accuracy levels of three gray system theory models for short-term traffic speed and travel time predictions. Gray models consistently demonstrate low prediction errors over all time series, thereby successfully improving the accuracy by approximately 50% on average in root-mean-squared errors and mean absolute percent errors. Traffic parameters can vary due to several factors, such as weather, accidents, and driving characteristics. Comert et al. [12] develop a model to predict traffic speed under abrupt change. The developed model is tested from 1-step to 45-step forecasts. The accuracy of predictions is improved until the 15-step forecast compared with nonadaptive and mean adaptive models. Although the developed model is not retrained on different data sets, the method provides better results than or close results to nonadaptive and adaptive models that are retrained on the corresponding data set. Chen and Rakha [13] develop an agent-based modeling approach to predict multistep ahead experienced travel time using real-time and historical spatiotemporal traffic data. The results show that the agent-based modeling approach produced the least prediction errors compared with other state-of-the-practice and state-of-the-art methods (such as instantaneous travel time, historical average, and k-nearest neighbor). The fast algorithm computation allows the proposed approach to be implemented in real-time applications in traffic management centers.

Papathanasopoulou et al. [14] develop a methodology to realize online calibration of microscopic traffic simulation models for dynamic multistep prediction of traffic measures. The application leads to less than 10% error in speed prediction even for 10 steps into the future in all considered data sets. Rajabzadeh et al. [15] provide a two-step approach based on stochastic differential equations to improve short-term prediction. At the first step, a Hull–White model is applied to obtain a baseline prediction model from previous days. Then, the extended Vasicek model is employed to model the difference between observations and baseline predictions (residuals) during an individual day. The results show that the proposed model can accurately input the missing data in traffic data set.

In summary, most studies focus on traffic state estimation and travel time reliability, which mainly emphasize the traffic quality but not the reliability of traffic quality. Advanced Traveler Information System is one of the functional areas of Intelligent Transportation Systems and aims to provide forecasted traffic information for travelers to make better

decisions. There are two methods to forecast the traffic data. One is one-step prediction, which predicts the reliability in the next one step. The other is fixed multistep prediction, in which the number of prediction steps is fixed. The results of fixed multistep prediction show that it easily leads to large prediction errors and the prediction errors gradually linearly increase with the number of multisteps [14]. For example, fixed multistep prediction can obtain the reliability in the next 10 days or even 30 days; however, the prediction error may be 10% in the next 10 days and 40% in the next 30 days. If an acceptable prediction errors is less than 20% for travelers (it can be obtained through investigation), do you want an accurate 10-day forecast or an inaccurate 30-day forecast? How to determine the number of predictable steps, which can not only guarantee exceeding the acceptable prediction errors but also provide as much future reliability data as possible. If there is a certain number S of steps to guarantee it, the number S of predictable steps is defined as the dynamic predictability.

The proposed concept of dynamic predictability needs to find the number S of predictable steps. After determining the number of predictable steps, the multistep prediction limited to the number of predictable steps will be performed. The realization of this theoretical principle is described in the following. Firstly, the traffic state reliability is defined, and the multistep forecasting theory and basic conditions are put forward in Section 2. Section 3 describes a two-phase model which determines the number of predictable steps in the upper-phase model and performs multistep prediction in the lower-phase model. Based on this, Section 4 verifies the validity and accuracy of the proposed method by applying actual traffic data. Section 5 summarizes the research conclusions.

2. Methodology

The definition of traffic state reliability (TSR) is the basis for setting a method to measure reliable level of traffic state. What is the relationship between past and present reliability and the future reliability? It is the discussion focus of this part.

2.1. Traffic State Reliability Calculation. For a link, intersection, or road network, TSR is defined as the degree of actual traffic flow state relative to the most reliable free flow state, which assumes that the reliability of free flow state is highest. It uses the TSR index to quantify evaluation, which is the ratio between actual speed and free flow speed in this research. TRS is classified to be equal to 1, when the actual speed is larger than or equal to the free flow speed and equal to $y^p(t)$ (see (1)) when the actual speed is less than the free flow speed. Therefore, TSR value is distributed from 0 to 1 and is calculated as

$$y^p(t) = \frac{v^p(t)}{v_f^p} \quad (1)$$

where $y^p(t)$ is the TSR of link or direction link p at the period t ; $v^p(t)$ is the average travel speed of link or direction link p at

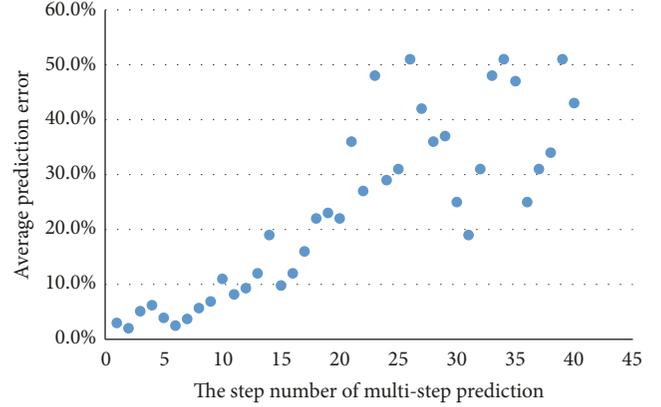


FIGURE 1: The relationship between the average prediction errors and the number S of steps.

the period t ; v_f^p is the free flow speed of link or direction link p .

2.2. Multistep Prediction Principle. One-step prediction [11] means predicting the reliability index in the next one step based on current and past reliability index series. Multistep prediction [12] means predicting the reliability index in the next several steps. The reliability series $\{y_{t-n+1}, \dots, y_{t-1}, y_t\}$ of Monday consists of TSR of the continuously $(t-n+1)$ th Monday, \dots , $(t-1)$ th Monday, \dots , t th Monday, also for all the other days of the week. The prediction principle of future S th series of reliability is

$$\hat{y}_{t+s} = f_s(y_{t-n+1}, \dots, y_{t-1}, y_t) \quad (2)$$

where $1 \leq s \leq S$, $0 < n \leq t$, and t is the current time interval; \hat{y}_{t+s} represents the predictive value of reliability in the next $(t+s)$ th interval; f_s represents the predictive function; n is the number of time intervals.

The above method can be used to offline predict the reliability of the next S steps by a kind of prediction method. The prediction results show that the average prediction errors gradually linearly increase with the number of steps. The graph shows the relationship between the average prediction errors and the number S of steps is seen in Figure 1 (Data Source in Section 4.1). The acceptable prediction errors investigated are less than 20% for travelers. Ultimately, the error 20% is the error threshold; the number S of steps corresponding to error 20% is the dynamic predictability. The prediction error is related to the characteristic of historical traffic flow data used. Traffic flow is characterized by randomness, regularity, and volatility according to traffic flow theory [8, 9]. Therefore, three key indexes should be first calculated to measure the characteristics of reliability series. Then, a model should be constructed to associate the key index with the dynamic predictability.

Fluctuation index is used to measure the random fluctuations of TSR series data. It refers to the ratio of variance to average reliability series. The less $B^p(t)$, the better the dynamic predictability. This index is expressed as

$$B^p(t) = \frac{A \cdot \sqrt{\sum_{i=1}^n ((y_i^{p2} - \bar{y}^{p2}) / (n-1))}}{\bar{y}^p}, \quad (3)$$

where n is the number of time intervals of TSR series. $B^p(t)$ is the volatility index of link or direction link p at period t . \bar{y}^p is the average value relative to n time interval. y_i^p is the TSR value of link or direction link p at the i th time interval A which is the adjustment coefficient.

Tendency index is used to quantitatively measure the tendency characteristic of the continuous increasing or decreasing of TSR series data. It refers to the cumulative amplitude change between two consecutive TSR of data series. The less $Q^p(t)$, the better the dynamic predictability.

This index is expressed as

$$Q^p(t) = \frac{|\sum_{i=1}^n ((y_i^p - y_i^p) / (n-k))|}{B \cdot k} \quad (4)$$

where $Q^p(t)$ is the tendency index of link or direction link p at period t . k is the gradient calculated based on the slope of TSR series. y_i^p is the reliability of link p at the current interval. B is the scaling factor.

Uncertainty characteristic index is used to quantitatively measure the disorder characteristic of the TSR series data. This index presents an inverse proportional relationship with the number of predictable steps. It can be quantitatively measured by a fuzzy entropy indicator based on fuzzy analysis theory. The less $L^p(t)$, the better the dynamic predictability. It is expressed as

$$L^p(t) = C \cdot \left(\log_2 n + \sum_{i=t-n+1}^t M_i^p \log_2 M_i^p \right), \quad (5)$$

where $L^p(t)$ is the fuzzy entropy. C is the scaling factor. M_i^p is the ambiguity and is expressed as

$$M_i^p = \frac{y_i^p}{\sum_{i=t-n+1}^t y_i^p} \quad (6)$$

Based on above description, it is required to find S first by using the dynamic predictability model to guarantee that the prediction error will not exceed the error threshold. Then, prediction within the maximum number S of steps will be conducted through the multistep forecasting model. Therefore, an upper-phase model is introduced to obtain the number S of predictable steps within the error threshold. After that, the multistep prediction limited to the number of predictable steps will be performed by the lower-phase model.

3. Study Model

Both the dynamic predictability and multistep prediction are nonlinear fitting problems. It is exactly the problem that wavelet neural network can solve. Wavelet neural network has the good capability of localization and nonlinear mapping [16]. Moreover, particle swarm optimization can further

improve the convergence rate and parameter iteration process. Therefore, a two-phase model of multistep prediction is designed based on wavelet neural network optimized by particle swarm algorithm. The upper-phase model is the dynamic predictability estimation method and the lower-phase model is the multistep prediction method.

3.1. Two-Phase Model of Multistep Prediction Based on Wavelet Neural Network Optimized by Particle Swarm Algorithm. A three-layer wavelet neural network is typically used to fit nonlinear correlations. This study uses a three-layer wavelet neural network to design the two-phase model for multistep prediction. First, in the upper-phase model, a three-layer wavelet neural network (Figure 1) is designed to estimate dynamic predictability. The inputs are the three key indexes of reliability series and the output is the number S of predictable steps. The number H of neurons in hidden layer is, respectively, set from 5 to 20 to achieve the fastest convergence rate. The nonlinear fitting is realized by using the Morlet function as shown in

$$\hat{S} = \sum_{h=1}^H \omega_h \psi \left(\frac{\sum_{i=1}^3 \omega_{hi} x_i - b_h}{a_h} \right) \quad (7)$$

where ψ is the Morlet wavelet function; ω_h is the weight between the output layer and the h th wavelet in the hidden layer; ω_{hi} is the weight between the h th wavelet in the hidden layer and the i th input; b_h and a_h are the translation factor and stretch factor of the h th wavelet in the hidden layer, respectively. The Morlet wavelet function is shown in

$$\psi(x) = \cos(1.75x) \exp\left(-\frac{x^2}{2}\right) \quad (8)$$

Second, in the lower-phase model, a three-layer wavelet neural network (Figure 2) is designed to carry out S steps prediction. Its input is reliability series data, and the output is S steps reliability in the future. The number of neurons in hidden layer is, respectively, set from 5 to 20 to achieve the fastest convergence rate. The hidden layer also uses the Morlet function as the excitation function, as shown in

$$\hat{y}_{t+s} = \sum_{f=1}^F \omega_{fs} \psi \left(\frac{\sum_{n=1}^N \omega_{fn} x_n - b_f}{a_f} \right) \quad (9)$$

where ψ is the Morlet wavelet function; ω_{fs} is the weight between the s th output and the f th wavelet in the hidden layer; ω_{fn} is the weight between the f th wavelet in the hidden layer and the n th input; b_f and a_f are the translation factor and stretch factor of the f th wavelet in the hidden layer, respectively.

The iterative optimization algorithm of wavelet neural network is algorithm with descent gradient. Notably, there is a local minimization problem. Particle swarm optimization can extract the transient characteristics of target as input, which has a better convergence and prediction accuracy. The principle of particle swarm optimization is to randomly generate a certain number of particle swarm optimizations,

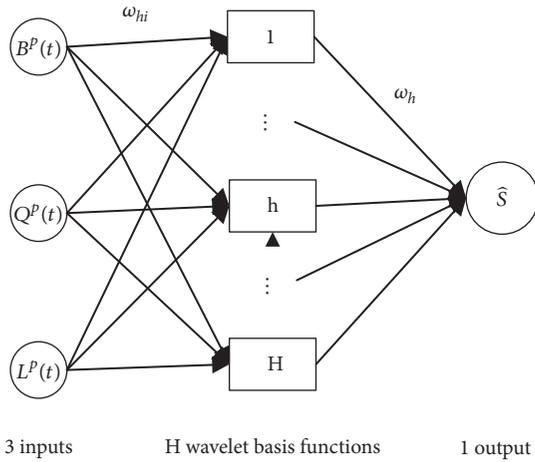


FIGURE 2: Structure of the wavelet neural network for upper phase.

which has three attribute indexes: position, velocity, and fitness. The position represents a possible solution to the optimization problem. The velocity is a vector, which determines the direction and size of iteration. The fitness is used to measure the good or poor position. If evaluation result is poor, then the position should be updated through iteration. Supposing the position vector of the m th particle is $X_m = (x_{m,1}, \dots, x_{m,d}, \dots, x_{m,D})$, then its velocity vector is $V_m = (v_{m,1}, \dots, v_{m,d}, \dots, v_{m,D})$, where D is the number of dimensions. During every iterative process, the m th particle first searches for its own best solution $pbest_m = (p_{m,1}, p_{m,2}, \dots, p_{m,D})$, that is, the individual extreme value. Then, all particles find the group best solution $gbest = (p_{g,1}, p_{g,2}, \dots, p_{g,D})$, that is, the global extreme value. When the fitness corresponding to the global extreme value satisfies the error requirement, the iteration reaches convergence and the global extreme value is obtained as the optimal solution. Otherwise, the velocity and position are iteratively updated according to the method shown in (10) and (11). The movement of particles can be appropriately restricted by setting the velocity range $[v_{\min}, v_{\max}]$ and the position range $[x_{\min}, x_{\max}]$ to ensure that the particles can efficiently search the global extreme value and avoid the local minimum problem.

$$v_{m,d}(i+1) = \omega \times v_{m,d}(i) + c_1 r_1 [p_{m,d} - x_{m,d}(i)] + c_2 r_2 [p_{g,d} - x_{m,d}(i)]; \quad (10)$$

$$x_{m,d}(i+1) = x_{m,d}(i) + v_{m,d}(i+1), \quad (11)$$

$$d = 1, 2, \dots, D,$$

where i is the current time of iterations. ω is the inertia factor. r_1 and r_2 are the random values evenly distributed between 0 and 1. c_1 and c_2 are the acceleration factors that are determined using the most commonly used penalty function strategy as shown in

$$c_1 = c_{1s} + \frac{c_{1e} - c_{1s}}{T_{\max}} \times Iter \quad (12)$$

$$c_2 = c_{2s} + \frac{c_{2e} - c_{2s}}{T_{\max}} \times Iter \quad (13)$$

The inertia factor ω controls the influences of iteration speed of previous step on the current iteration speed. Small ω is beneficial to convergence, whereas large ω is conducive to jump out of the local minimum for particle. Considering the global searching and local searching, the inertia factor ω is iterated by the linear decreasing method as shown in

$$\omega_{Iter} = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{T_{\max}} \times Iter, \quad (14)$$

where $Iter$ is the current number of iterations, $\omega_{\max} = 0.95$, $\omega_{\min} = 0.25$, T_{\max} is the total times of iterations, $C_{1s} = 2.75$, $C_{1e} = 1.25$, $C_{2s} = 0.25$, and $C_{2e} = 2.25$.

3.2. Algorithm of Two-Phase Model for Multistep Prediction. For a particular place, the current reliability series is $\{y_{t-n+1}, \dots, y_{t-1}, y_t\}$ at the time t and three key indexes are $B(t), Q(t), L(t)$. G data samples of historical series $Y_1, \dots, Y_g, \dots, Y_G$ are available in database, where $Y_g = \{y_g^{t-n+1}, \dots, y_g^{t-1}, y_g^t\}$.

Step 1. The three key indexes $T_1, \dots, T_g, \dots, T_G$ of data samples are calculated, where $T_g = \{B_g(t), Q_g(t), L_g(t)\}$, and the fixed X steps ($X \gg S$) prediction of data samples is conducted by applying the wavelet neural network optimized by particle swarm algorithm. Then, the actual number $S_1, \dots, S_g, \dots, S_G$ of predictable steps is determined by combining with an acceptable error threshold, in which the prediction error is continuously less than the threshold. Among the data samples in the database, two-thirds of the sample data (total I) is used as the training data set, and the rest is used as the testing data set.

Step 2. The upper-phase model is constructed as shown in Figure 2. The number H of neurons in hidden layer is set from 5 to 20 in training the network for selecting the optimal number of neurons.

Step 3. The training parameter is optimized by particle swarm algorithm. M particles are randomly generated. The position vector of each particle is represented as the real number coded that includes the stretch factor a_h , translation factor b_h of initialized wavelet function, the connection weight ω_{ih} between the i th input and the h th hidden layer, and the connection weight ω_h between the h th hidden layer and output layers. The function is expressed as

$$X_m = (x_{m,1}, \dots, x_{m,d}, \dots, x_{m,D}) = (a_1, \dots, a_H, b_1, \dots, b_H, \omega_{11}, \dots, \omega_{1H}, \omega_{21}, \dots, \omega_{2H}, \omega_{31}, \dots, \omega_{3H}, \omega_1, \dots, \omega_H) \quad (15)$$

where $D = 3 \times H + H \times 1 + 2 \times H = 6H$.

Step 4. The upper-phase model is trained by using the data sample from the training data set. The three key indexes $B_i(t), Q_i(t), L_i(t)$ of data sample are the input and the number S of predictable steps is the desired output. The **SAP**E (average prediction error of steps) between the predicted and desired output are used as the fitness value of particle as shown in

$$SAP\!E = \frac{\sum_{i=1}^I |S_i - \hat{S}_i|}{I \times S_i} \times 100\%. \quad (16)$$

where $SAP\!E$ is the relative average prediction error of steps. S_i is the actual number of predictable steps of the i th data sample. \hat{S}_i is the predicted number of predictable steps of the i th data sample.

Step 5. The upper-phase model is tested using the data sample from the testing data set until the average relative error satisfies the requirements that means the upper-phase model fits well. Then, the predictable step number S of the reliability series at the current period t is forecasted through input $B(t), Q(t), L(t)$. Otherwise, Step 3 is repeated.

Step 6. After obtaining the number S of predictable steps, a multistep prediction model is constructed to forecast the reliability in the subsequent S step based on wavelet neural network optimized by particle swarm algorithm. The actual value $Y'_1, \dots, Y'_g, \dots, Y'_G$ of reliability data sample in the future S steps is extracted from the database, where $Y'_g = \{y_g^{t+1}, \dots, y_g^{t+S-1}, y_g^{t+S}\}$. Two-thirds of the sample data (total I) are used as the training data set, and the rest are used as the testing data set.

Step 7. The lower-phase model is constructed as shown in Figure 3. The number F of neurons in hidden layer is set from 5 to 20 in training the network for selecting the optimal number of neurons.

Step 8. The training parameter is optimized using particle swarm algorithm and M particles are randomly generated. The position vector of each particle is represented by the real number coded that includes the stretch factor a_f , translation factor b_f of initialized wavelet function, the connection weight ω_{fn} between the n th input and the f th hidden layer, and the connection weight ω_{fs} between the f th hidden layer and the s th output layer. The function is expressed as

$$X_m = (x_{m,1}, \dots, x_{m,d}, \dots, x_{m,D}) = (a_1, \dots, a_F, b_1, \dots, b_F, \omega_{11}^r, \dots, \omega_{1F}^r, \dots, \omega_{NF}^r, \omega_{11}^c, \dots, \omega_{FS}^c) \quad (17)$$

where $D = 2 \times F + N \times F + F \times S = (2 + N + S)F$.

Step 9. The lower-phase model is trained by using the data sample from the training data set. The reliability series data $y_g^{t-n+1}, \dots, y_g^{t-1}, y_g^t$ are the input and the actual value $y_g^{t+1}, \dots, y_g^{t+S-1}, y_g^{t+S}$ in the future S steps is the desired output. The **MAPE** (average predictive error of multistep) between

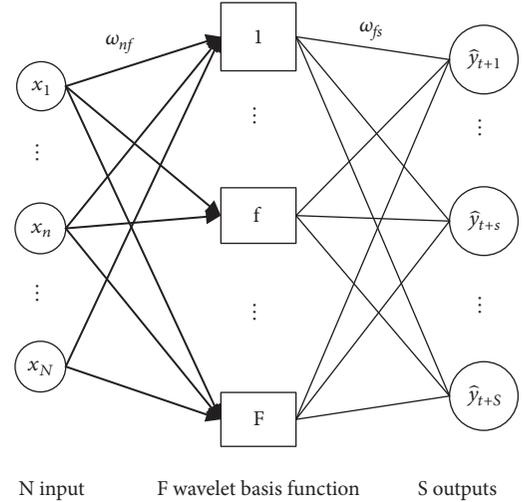


FIGURE 3: Structure of the wavelet neural network for lower phase.

the predicted and desired output is used as the fitness value of particle as shown in

$$MAPE = \frac{1}{I \times S} \sum_{s=1}^S \sum_{i=1}^I \frac{|y_i^s - \hat{y}_i^s|}{y_i^s} \times 100\%, \quad (18)$$

where $MAPE$ is the average predictive error of multistep prediction. y_i^s is the actual reliability of the i th data sample in the future $t+s$ time interval. \hat{y}_i^s is the predicted reliability of the i th data sample in the future $t+s$ time interval. I is the total number of data samples in training data set. S is the number of predictable steps.

Step 10. The lower-phase model is tested using the data sample from the testing data set. If the average relative error $MAPE$ satisfies the requirements, then the training is terminated and the reliability in future S steps of reliability series at the current period t is forecasted. Otherwise, Step 8 is repeated.

In order to verify the effectiveness of the above model and algorithm, it is necessary to collect and analyze the actual traffic flow parameters, calculate proposed TSR, estimate the dynamic predictability, and predict the multistep reliability in future. Eventually, it is necessary to compare the research result with actual traffic state to confirm the validity of the proposed method.

4. Case Study

The empirical data were obtained from a freeway of Shanghai. The structure of road network is shown in Figure 4. The process of empirical analysis is divided into three steps: data source, experimental program, experimental results, and comparative analysis.

4.1. Data Source. Traffic data was collected by the loop detector from a freeway in Shanghai of China. The collected

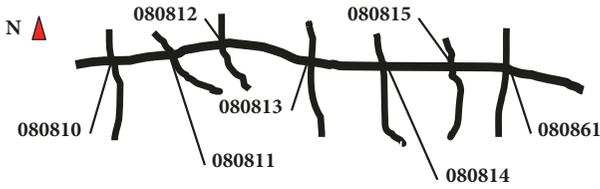


FIGURE 4: Diagram of the road experimental area.

traffic parameters mainly include speed, traffic volume, and occupancy. The size of data set is 24 hours of every day from May to July. The time interval of collected data is 1 minute.

4.2. Experimental Program. The reliability of each link was calculated using (1). The data collected in May and June is used as training data set, and the data collected in July is used as testing data set. Parameters A, B, and C have been set according to analysis results of sensitivity analysis method through assigning 5 to 200, respectively. The results show that A is set as 50, B is set as 5, and C is set as 150.

First, the step number of iterations is used to measure convergence effect of wavelet neural network optimized by particle swarm algorithm. Second, the mean relative error SAPE is calculated to measure performance of the upper-phase model. Third, the mean relative error MAPE is calculated to measure performance of the lower-phase model. As traffic state on weekday is usually very different from that on weekend, the number of neurons in the middle layer is separately determined, and the forecasting work is also verified separately.

Liu et al. [17] conducted a prediction model of time series based on backpropagation neural network. Bao et al. [18] performed a multistep-ahead time series prediction using multiple-output support vector regression. The proposed method is verified comparatively with backpropagation neural network and support vector regression.

The difference between wavelet neural network and classical artificial neural network (such as backpropagation neural network) is excitation function in middle hidden layer. The excitation function of the latter usually uses sigmoid function, which leads to nonconvergence or slow convergence. However, the wavelet neural network adopts wavelet function as excitation function. It introduces translation scale and scaling factor to extract local information. Thus, the convergence speed, approximation precision, and performance are improved compared with those of classical artificial neural network.

Support vector machine is one learning method for small sample. It ignores probability measure and law of large numbers unlike existing statistical methods. It is also based on the minimization principle of structural risk. The global optimal solution is obtained through the optimization problem of convex quadratic, which has high generalization capability and an advantage in terms of generalization and classification.

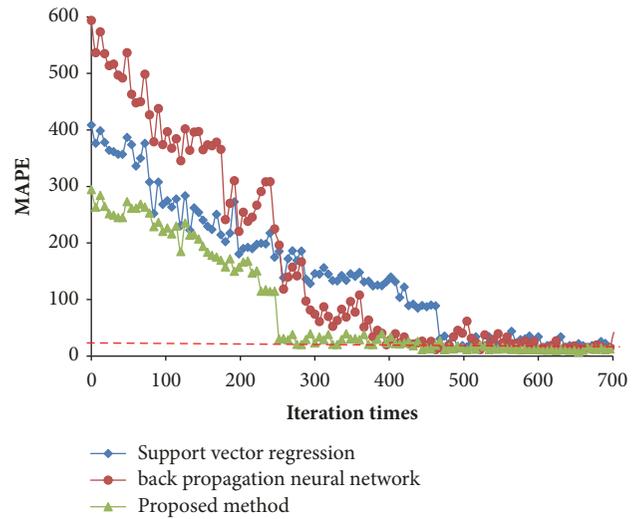


FIGURE 5: Convergence of upper-phase model based on different methods.

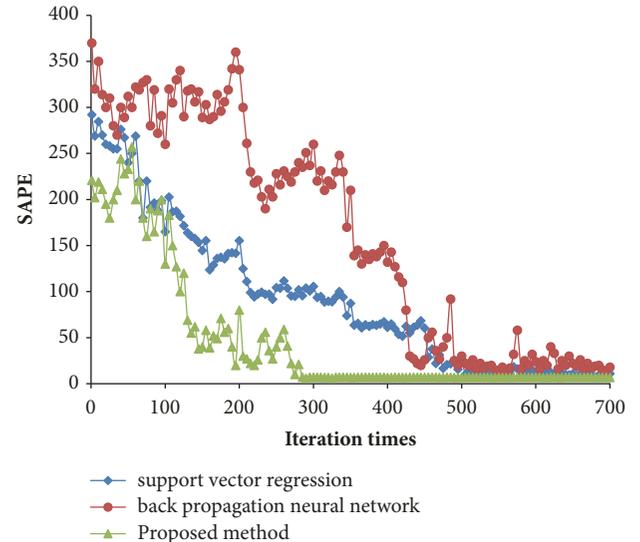


FIGURE 6: Convergence of lower-phase model based on different methods.

4.3. Experimental Results and Comparative Analysis

4.3.1. Convergence Analysis of Wavelet Neural Network Optimized by Particle Swarm Algorithm. The convergence rate was trained and comparatively analyzed in upper-phase and lower-phase model using the same set of reliability data series and under the same conditions. The result is shown in Figures 5 and 6.

Figures 5 and 6 show that upper-phase model based on wavelet neural network optimized by particle swarm algorithm begins to converge at 256 seconds, whereas backpropagation neural network and support vector regression achieve convergence effect at 398 and 463 seconds, respectively. The lower-phase model based on wavelet neural network optimized by particle swarm algorithm begins to converge at

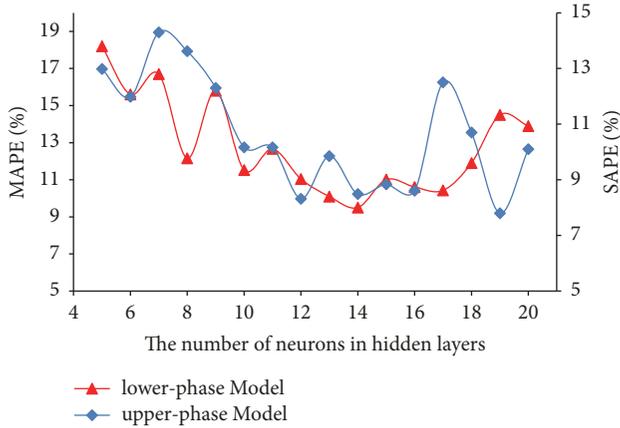


FIGURE 7: Relationship between MAPE/SAPE and the number of neurons on weekday.

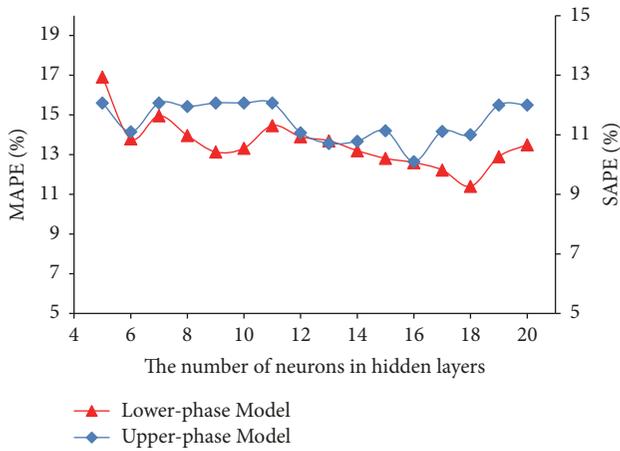


FIGURE 8: Relationship between MAPE/SAPE and the number of neurons on weekend.

291 seconds, whereas the other begins convergence effect at 487 and 526 seconds, respectively. Therefore, the parameter-optimizing process based on particle swarm algorithm contributes greatly to the fast convergence speed.

4.3.2. *Determining the Number of Neurons, Respectively, Based on Weekday and Weekend Data.* In order to determine the number of neurons in hidden layers of upper-phase and lower-phase model, the number of neurons is set from 5 to 20 to train, respectively, based on weekday and weekend data. The result is shown in Figures 7 and 8.

Based on weekday data, Figure 7 shows that SAPE was lowest when the number of neurons in hidden layer is 12. MAPE is lowest when the number of neurons in hidden layer is 14. Meanwhile, based on weekend data, Figure 8 shows that the SAPE is lowest when the number of neurons in hidden layer was 16. Meanwhile, the MAPE is lowest when the number of neurons in the hidden layer is 18.

4.3.3. *Performance of Upper-Phase Model, Respectively, Based on Weekday and Weekend Data.* Based on weekday data,

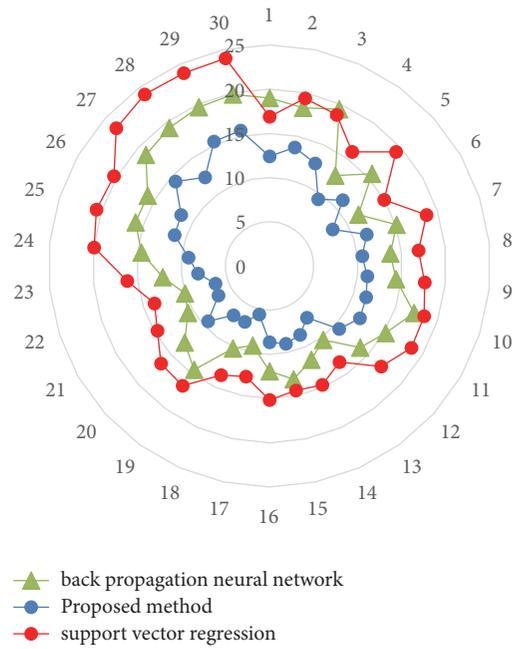


FIGURE 9: Prediction results of 30 steps for weekday reliability based on different methods.

the number of predictable steps is estimated and analyzed comparatively based on wavelet neural network optimized by particle swarm algorithm, the backpropagation neural network, and the support vector regression. The result is shown in Table 1. It shows that the proposed method presents a better performance compared with referenced methods. Meanwhile, based on weekend data, the number of predictable steps is estimated and analyzed comparatively. The result is shown in Table 2, which shows that the proposed method presents a better estimation performance compared with referenced methods.

4.4. *Performance of Lower-Phase Model, Respectively, Based on Weekday and Weekend Data.* Based on weekday data, the multistep prediction is analyzed comparatively based on wavelet neural network optimized by particle swarm algorithm, the backpropagation neural network, and the support vector regression. The result is shown in Table 3. It shows that MAPE of lower-phase model is less than that of referenced methods. Meanwhile, based on weekend data, the multistep prediction result is shown in Table 4. It shows that the MAPE of lower-phase model is the lowest, 12.01%.

Moreover, 30 steps in future are predicted, respectively, based on weekday and weekend data. The result is shown in Figures 9 and 10.

Figures 9 and 10 indicate that prediction error increases as the increase of step number which presents a positive correlation. Meanwhile, MAPE of the proposed method is less than that of referenced methods. Moreover, the prediction performance based on weekday data is better than that of weekend.

Meanwhile, 24-step reliability is predicted, respectively, on Monday and Saturday for the direction link 080810 which

TABLE 1: SAPE of proposed method and referenced method for upper-phase model based on weekday data (%).

Reliability index	Calibration data set	support vector regression	back propagation neural network	Proposed method
reliability series	training sample	15.41	19.87	7.92
	testing sample	20.75	20.74	8.63

TABLE 2: SAPE of proposed method and referenced method for upper-phase model based on weekend data (%).

Reliability index	Calibration data set	support vector regression	back propagation neural network	Proposed method
reliability series	training sample	16.51	22.91	10.07
	testing sample	24.73	20.62	12.73

TABLE 3: MAPE of proposed method and referenced method for lower-phase model based on weekday data (%).

Reliability index	Data set	support vector regression	back propagation neural network	Proposed method
reliability series	training sample	17.92	20.7	8.91
	testing sample	20.15	21.58	10.02

TABLE 4: MAPE of proposed method and referenced method for lower-phase model based on weekend data (%).

Reliability index	Data set	support vector regression	back propagation neural network	Proposed method
TSR series	training sample	22.32	23.89	12.01
	testing sample	26.13	23.91	13.54

is from south to north in the intersection. The forecasting period is from 08:00:00 to 08:23:00, and the time interval is 1 min. The result is shown in Figures 11 and 12, which shows that the reliability predicted by the proposed method is closer to actual values than that of backpropagation neural network and support vector regression. The proposed method obtains the lowest average relative error. Therefore, the proposed method shows a better prediction performance.

5. Conclusions

In view of the fact that the traffic state should be a continuous variable, this study proposes the concept of TSR to measure the stability of actual traffic flow state relative to the most reliable free flow state. In order to know TSR in future, a two-phase model of multistep prediction has been designed based on wavelet neural network optimized by particle swarm algorithm. In the upper-phase model, the number S of predictable steps within the error threshold is determined according to three key indexes that are fluctuation index, tendency index, and uncertainty index. In the lower-phase model, a three-layer wavelet neural network has been designed to carry out prediction in S steps. In two-phase model, training parameter has been optimized by particle swarm algorithm, which has a better convergence and prediction accuracy. The proposed model can guarantee not only forecasting within the acceptable prediction errors but also providing as much future reliability data as possible. Moreover, it aggregates the prevailing nonlinear mapping capability of wavelet neural network and the excellent parameter optimization

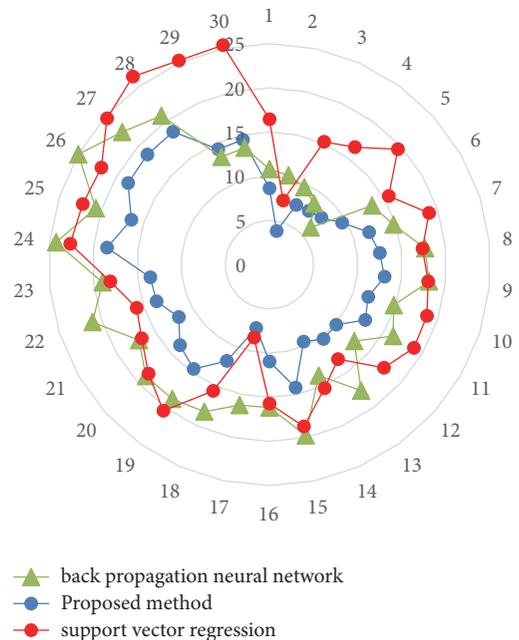


FIGURE 10: Prediction results of 30 steps for weekend reliability based on different methods.

effect of particle swarm algorithm. The empirical analysis shows that parameter-optimizing process based on particle swarm algorithm has fast convergence speed. The proposed method presents a better prediction performance compared with backpropagation neural network and support vector

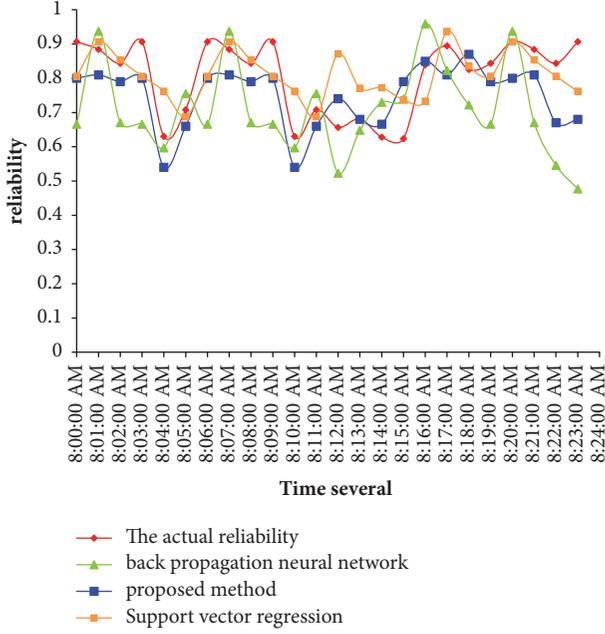


FIGURE 11: Multistep prediction of weekday reliability based on different methods.

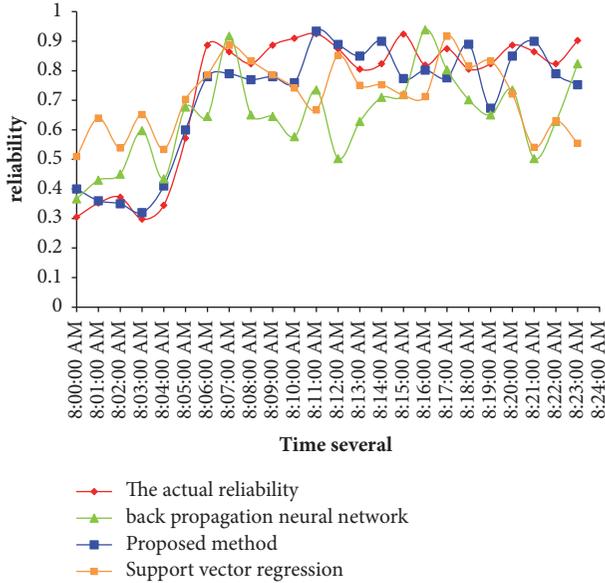


FIGURE 12: Multistep prediction of weekend reliability based on different methods.

regression. Moreover, the prediction performance of weekday is better than that of weekend.

Notation

The Symbols Used in This Study

- A : The adjustment coefficient
 B : The scaling factor
 $B^P(t)$: The volatility index of link or direction link p at period t

- C : The scaling factor
 D : The number of dimensions
 I : The total number of data samples in training data set
 $L^P(t)$: The fuzzy entropy
 $MAPE$: The average predictive error of multistep prediction
 M_i^P : The ambiguity
 N : The number of continuous weeks
 $Q^P(t)$: The tendency index of link or direction link p at period t
 S : The number of predictable steps
 $SAPe$: The relative average predictive error of steps
 S_i : The actual number of predictable steps of the i th data sample
 \hat{S}_i : The predicted number of predictable steps of the i th data sample
 $S_1, \dots, S_g, \dots, S_G$: The actual number of predictable steps of data samples of historical reliability series
 T_{max} : The total number of iterations
 $TSR^P(t)$: Reliability of link or direction link p at the period t
 $T_1, \dots, T_g, \dots, T_G$: the three key indexes, where $T_g = \{B_g(t), Q_g(t), L_g(t)\}$
 $V_m = (v_{m,1}, \dots, v_{m,d}, \dots, v_{m,D})$: The velocity vector of the m th particle
 $X_m = (x_{m,1}, \dots, x_{m,d}, \dots, x_{m,D})$: The position vector of the m th particle
 $Y_1, \dots, Y_g, \dots, Y_G$: G data samples of historical reliability series, where $Y_g = \{y_g^{t-n+1}, \dots, y_g^{t-1}, y_g^t\}$
 $Y'_1, \dots, Y'_g, \dots, Y'_G$: The actual value of reliability data sample in the future S steps from database, where $Y'_g = \{y_g^{t+1}, \dots, y_g^{t+S-1}, y_g^{t+S}\}$
 a_f : The stretch factor of the f th wavelet in the hidden layer of the lower-phase model
 a_h : The stretch factor of the h th wavelet in the hidden layer of the upper-phase model

b_f :	The translation factor of the f th wavelet in the hidden layer of the lower-phase model
b_h :	The translation factor of the h th wavelet in the hidden layer of the upper-phase model
c_1 :	The acceleration factor
c_2 :	The acceleration factor
f_s :	The predictive function— n is the number of time intervals
$g_{best} = (p_{g,1}, p_{g,2}, \dots, p_{g,D})$:	The best solution for the group obtained by all particles, that is, the global extreme value
i :	The current times of iterations
k :	The gradient calculated based on the slope of reliability series data— n is the number of time intervals of reliability series
n :	The number of time intervals of reliability series
$p_{best}_m = (p_{m,1}, p_{m,2}, \dots, p_{m,D})$:	The best solution of the m th particle, that is, the individual extreme value
r_1 and r_2 :	A random value evenly distributed between 0 and 1
t :	The current time interval
\hat{y}_{t+s} :	The predictive value of reliability in the next $(t + s)$ th interval
$\{y_{t-n+1}, \dots, y_{t-1}, y_t\}$:	The historical series of reliability
y_i^s :	The actual reliability of the i th data sample in the future $t+s$ time interval
\hat{y}_i^s :	The predicted reliability of the i th data sample in the future $t+s$ time interval
\bar{y}^p :	The average value of reliability series of link or direction link p at n time interval
y_i^p :	The reliability value of link or direction link p at the i th time interval
y_t^p :	The reliability of link p at the current interval
y_j^p :	The j th reliability in reliability series of link p
ω :	The inertia factor

$\omega_{kj}^p(t)$:	The traffic state values of link or direction link p at the period t on the j th day of the k th week
ω_h :	The weight between the output layer and the h th wavelet in the hidden layer of the upper-phase model
ω_{hi} :	The weight between the h th wavelet in the hidden layer and the i th input of the upper-phase model
ω_{fs} :	The weight between the s th output and the f th wavelet in the hidden layer of the lower-phase model
ω_{fn} :	The weight between the f th wavelet in the hidden layer and the n th input of the lower-phase model
ψ :	The Morlet wavelet function.

Conflicts of Interest

The authors (Jufen Yang, Zhigang Liu, and Guiyan Jiang) declare that there are no conflicts of interest regarding the publication of this paper. They confirm that the mentioned funding in the Acknowledgments did not lead to any conflicts of interest.

Acknowledgments

The authors greatly appreciate the support provided by the National Key Research and Development Plan of China (Grant no. 2017YFC0804900), the National Natural Science Foundation of China (Grant no. 71701124 and Grant no. 51278257), and the Special Fund of Training Scheme for Young Teachers of Universities in Shanghai (Grant no. ZZGCD15116).

Supplementary Materials

The flow chart of the two-phase model of multistep prediction based on wavelet neural network optimized by particle swarm algorithm is shown in Supplementary Materials as an annex. (*Supplementary Materials*)

References

- [1] C. Antoniou, H. N. Koutsopoulos, and G. Yannis, "Dynamic data-driven local traffic state estimation and prediction," *Transportation Research Part C: Emerging Technologies*, vol. 34, pp. 89–107, 2013.
- [2] G. Hiribarren and J. C. Herrera, "Real time traffic states estimation on arterials based on trajectory data," *Transportation Research Part B: Methodological*, vol. 69, pp. 19–30, 2014.
- [3] L. Mannini, S. Carrese, E. Cipriani, and U. Crisalli, "On the short-term prediction of traffic state: An application on urban freeways in ROME," *Transportation Research Procedia*, vol. 10, pp. 176–185, 2015.
- [4] A. Nantes, D. Ngoduy, A. Bhaskar, M. Miska, and E. Chung, "Real-time traffic state estimation in urban corridors from heterogeneous data," *Transportation Research Part C: Emerging Technologies*, vol. 66, pp. 99–118, 2016.

- [5] Z. Zheng and D. Su, "Traffic state estimation through compressed sensing and Markov random field," *Transportation Research Part B: Methodological*, vol. 91, pp. 525–554, 2016.
- [6] X. Kong, Z. Xu, G. Shen, J. Wang, Q. Yang, and B. Zhang, "Urban traffic congestion estimation and prediction based on floating car trajectory data," *Future Generation Computer Systems*, vol. 61, pp. 97–107, 2016.
- [7] D. Wang, H. Qi, and C. Xu, "Reviewing Traffic Reliability Research," *Journal of Transportation Systems Engineering and Information Technology*, vol. 10, no. 5, pp. 12–21, 2010.
- [8] Y. Xiao, N. Coulombel, and A. D. Palma, "The valuation of travel time reliability: does congestion matter?" *Transportation Research Part B: Methodological*, vol. 97, pp. 113–141, 2017.
- [9] X. Xiao, J. Yang, S. Mao, and J. Wen, "An improved seasonal rolling grey forecasting model using a cycle truncation accumulated generating operation for traffic flow," *Applied Mathematical Modelling: Simulation and Computation for Engineering and Environmental Systems*, vol. 51, pp. 386–404, 2017.
- [10] F. He, X. Yan, Y. Liu, and L. Ma, "A Traffic Congestion Assessment Method for Urban Road Networks Based on Speed Performance Index," *Procedia Engineering*, vol. 137, pp. 425–433, 2016.
- [11] A. Bezuglov and G. Comert, "Short-term freeway traffic parameter prediction: Application of grey system theory models," *Expert Systems with Applications*, vol. 62, pp. 284–292, 2016.
- [12] G. Comert, A. Bezuglov, and M. Cetin, "Adaptive traffic parameter prediction: Effect of number of states and transferability of models," *Transportation Research Part C: Emerging Technologies*, vol. 72, pp. 202–224, 2016.
- [13] H. Chen and H. A. Rakha, "Multi-step prediction of experienced travel times using agent-based modeling," *Transportation Research Part C: Emerging Technologies*, vol. 71, pp. 108–121, 2016.
- [14] V. Papathanasopoulou, I. Markou, and C. Antoniou, "Online calibration for microscopic traffic simulation and dynamic multi-step prediction of traffic speed," *Transportation Research Part C: Emerging Technologies*, vol. 68, pp. 144–159, 2016.
- [15] Y. Rajabzadeh, A. H. Rezaie, and H. Amindavar, "Short-term traffic flow prediction using time-varying Vasicek model," *Transportation Research Part C: Emerging Technologies*, vol. 74, pp. 168–181, 2017.
- [16] H.-J. Yang and X. Hu, "Wavelet neural network with improved genetic algorithm for traffic flow time series prediction," *Optik - International Journal for Light and Electron Optics*, vol. 127, no. 19, pp. 8103–8110, 2016.
- [17] Y.-K. Liu, F. Xie, C.-L. Xie, M.-J. Peng, G.-H. Wu, and H. Xia, "Prediction of time series of NPP operating parameters using dynamic model based on BP neural network," *Annals of Nuclear Energy*, vol. 85, pp. 566–575, 2015.
- [18] Y. Bao, T. Xiong, and Z. Hu, "Multi-step-ahead time series prediction using multiple-output support vector regression," *Neurocomputing*, vol. 129, pp. 482–493, 2014.



Hindawi

Submit your manuscripts at
www.hindawi.com

