Journal of Highway and Transportation Research and Development
2024, 18(4) : 34−41

Research Article
ISSN 2095−6215
DOI: 10.26599/HTRD.2024.9480030

Short term traffic flow prediction and timing optimization at signalized intersections based on SG-LSTM and particle swarm optimization

Lei Yang*, Ruijun Guo**

School of Transportation Engineering, Dalian Jiaotong University , Dalian, Liaoning 116023, China

Abstract: In addressing the challenge of spatio-temporal correlation in traffic flow at signalized intersections, this study introduces an innovative methodology that integrates a SG-LSTM neural network with a particle swarm optimization algorithm. The proposed methodology involves the pre-processing of original traffic data to enhance its predictability, followed by the development of a short-term traffic flow prediction model utilizing the LSTM neural network. Utilizing the outcomes of the traffic flow predictions, a dynamic timing model for signalized intersections is formulated through the application of the particle swarm optimization algorithm. Furthermore, a multi-objective optimization model is established to refine the timing scheme of the signalized intersections, incorporating constraints that consider the maximization of overall intersection capacity, minimization of average delay, and reduction of average queuing times. The Pareto compromise programming method is employed to perform dimensionless processing of the three performance indicators, while the fuzzy preference method is utilized to ascertain the weight relationships among the objective functions. The optimized signal timing scheme is derived through the implementation of the particle swarm optimization algorithm in Matlab. Experimental results indicate that the proposed methodology surpasses the traditional Webster model in terms of performance indicators, thereby affirming the effectiveness and reliability of the approach.

Keywords: traffic engineering; short-term traffic flow prediction; multi-objective optimization; SG-LSTM; particle swarm optimization; signalized intersections

1 Introduction

Accurate and timely traffic flow prediction can provide advanced traffic flow information for signal timing optimization and realize the function of determining timing plan in advance to adapt to the traffic scene at the intersection in the next stage. Due to the temporal nature of traffic flow at signalized intersections, short-time traffic flow prediction methods are commonly used to forecast traffic flow. The research results of short-time traffic flow prediction can be summarized into three categories: one is the model based on statistical theory, such as integrated moving average autoregressive model (ARIMA), Kalman filter model, etc. The other is the model based on artificial intelligence. Zhong et al.[1] proposed a short-term traffic flow prediction model for signalized intersections based on long short-term memory (LSTM), whose prediction effect is better than that of the traditional BP neural network model. Mou et al.[2] proposed a time information increment LSTM (T-LSTM) prediction model to predict the traffic flow of a section, which can capture the internal relationship between traffic flow and time information to improve the prediction accuracy. The third is the Mixture model, which integrates advanced technologies such as multiple optimization algorithms into a prediction model to improve prediction performance. Vlahogianni et al.[3] combined genetic algorithms with multi-layer structural optimization to help select appropriate neural network structures for traffic flow data with temporal and spatial characteristics. Long et al.[4] proposed a short-term passenger flow prediction model for urban rail transit (DBN-P/GSVM) based on deep belief network (DBN) and SVM.

By controlling intersection signal lights, traffic congestion can be alleviated, traffic capacity can be enhanced,

Received: 28 July 2024; **Revised:** 12 August 2024; **Accepted:** 28 December 2024

* *E-mail address*: yl14242@163.com

** *Corresponding author*: rjguo@163.com

vehicle delays can be reduced, and traffic flow balance can have a significant effect on improving the operation efficiency of urban road network[5]. Chou et al.[6] used fuzzy logic for signal control at intersections, which can adapt to multi-intersections and multi-lanes and achieve low delays under various traffic loads. Xin et al.[7] proposed a genetic algorithm-based signal coordination control method for the main road at adjacent intersections, which effectively reduced vehicle delays. Cao[8] used the neural network to predict the short-term flow of each inlet road at the intersection, and used the simulated annealing algorithm to build the intersection dynamic timing model, and the simulation results proved the effectiveness of the model. On the basis of short-term traffic flow prediction, Dai[9] established a multi-objective optimization model aiming at average vehicle delay, capacity and carbon monoxide emission at intersections.

Traffic flow prediction and signal timing at signalized intersection are mostly studied separately. Short-term traffic flow prediction methods, such as LSTM and DBN, are generally used to predict traffic flow at signalized intersections.The method of multi-objective optimization of signal timing is often used, which can comprehensively consider the specific situation of the intersection and is more widely used.It is necessary to provide drivers with advanced traffic conditions through accurate traffic flow prediction data for signal timing.As short-term traffic flow prediction requires a large amount of traffic data support, if the data quality is poor or the data is missing, the prediction accuracy will be seriously affected. At present, most studies directly use traffic flow data as a training set for prediction, and the prediction accuracy needs to be improved.In this paper, an LSTM neural network signal intersection prediction method based on S-G filter is proposed. On the basis of the prediction results, particle swarm optimization is used to solve the multi-objective optimization model to optimize the signal timing of the intersection. The comparison of the optimization results shows that the method has good performance.

2 SG-LSTM neural network model

2.1 Savitzky-golay filter

Savitzky-Golay filter (often abbreviated as S-G filter) is a filtering method based on local polynomial least square fitting in the time domain.It can be applied to a set of data to smooth the data, which can improve the accuracy of the data without changing the signal trend and width. Its advantage is that on the same curve, different widths can be arbitrarily selected at any position to meet the needs of different smoothing and filtering. Especially, when processing time sequence data such as traffic flow, it has obvious advantages for sequence processing at different stages. The S-G filter can reduce the noise interference of data, improve the periodicity of data and predict the traffic flow more accurately. Savitzky-Golay smoothing formula is shown in Equation (1) [10].

$$\overline{x}_{\xi} = \sum_{r=\frac{1-m}{2}}^{\frac{m-1}{2}} c_{\xi} x_{r+\xi}, \tag{1}$$

where, $\overline{x}_{\xi}$ for smooth result; m is the filter window size; $x_{r+\xi}$ for the data points in the window; c_{ξ} for smoothing coefficient.

The least square method with m smoothing coefficients is used to estimate the $x_{r+\xi}(\xi = 1, \cdots, T)$, window size affects the effect of the S-G filter.

Calculation of smoothing coefficient C_{ξ}: The adjacent data points $X = \left\{x_{\frac{1-m}{2}}, \cdots, x_0, \cdots, x_{\frac{m-1}{2}}\right\}$ and $Z = \left\{\frac{1-m}{2}, \cdots, 0, \frac{m-1}{2}\right\}$ were fitted by linear least squares method, which was an integer. The coefficients include a_0, a_1. And so on using linear least square method. The calculation formula is shown in Equation (2) to (5).

$$x_r = a_0 + a_1 z_r + a_2 z_r^2 + \cdots + a_k - z_r^{k-1}, \tag{2}$$

where, x_r is the data output after fitting; z_r is the data to be fitted; a Parameter to be solved.

x_r has m equations, and the matrix $\boldsymbol{X}$ is expressed as:

$$\boldsymbol{X} = \boldsymbol{Z} \cdot \boldsymbol{A}, \tag{3}$$

where, $\boldsymbol{X} \in R^{m*1}$, $\boldsymbol{Z} \in R^{m*k}$ and $\boldsymbol{A} \in R^{k*1}$; A is determined by the least squares fitting method.

Then use the least square method to calculate $\overline{\boldsymbol{A}}$:

$$\overline{\boldsymbol{A}} = (\boldsymbol{Z}^{\mathrm{T}} \cdot \boldsymbol{Z})^{-1} \cdot \boldsymbol{Z}^{\mathrm{T}} \cdot \boldsymbol{X}. \tag{4}$$

Finally, the smoothing coefficient matrix $\boldsymbol{C}_{\xi}$ of $\boldsymbol{X}$ can be written:

$$\boldsymbol{C}_{\xi} = (\boldsymbol{Z}^{\mathrm{T}} \cdot \boldsymbol{Z})^{-1} \cdot \boldsymbol{Z}^{\mathrm{T}}. \tag{5}$$

2.2 LSTM neural network

Long Short-Term Memory Network (LSTM) has a memory like a computer, from which information can be read, stored, deleted and written, so that the neural network can have the function of long-term memory without losing information[11]. Think of this memory as a gated unit, and a gated switch can decide whether the unit retains information or not. Through training, the network can learn which information is important and which is not, thus improving the accuracy of prediction. LSTM neural network highlights the concept of time sequence of data and can process data with time series, which is very suitable for short-term traffic flow prediction.

LSTM neural network has the same chain structure as Recurrent Neural Network (RNN), but the internal structure of the cell between them is different. RNN neural network has no cell state, and its activation function is only tanh function. LSTM neural network can remember by cell state, and it also has input gate, forgetting gate, output gate, and the introduction of sigmoid function and tanh function combination, add summation operation to solve the problem of gradient explosion and gradient disappearance. As shown in Figure 1, the network framework of the LSTM neural network model is given. The calculation process is as follows: First, the hidden layer of the previous time is used to output h_{t-1} and the input x_t of the current time, and the three gates and candidate states are calculated. The memory unit is updated by combining the forgetting gate and the input gate with the output gate, and the information of the internal state is transmitted to the external state.

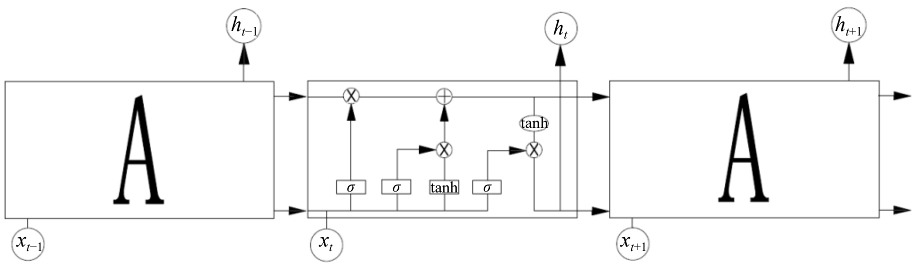

Fig. 1 LSTM neural network model framework

The calculation formulas of forgetting gate, input gate and output gate are as follows:

$$f_t = \sigma(\boldsymbol{W}_\text{f} x_t + \boldsymbol{U}_\text{f} h_{t-1} + b_\text{f}), \tag{6}$$

$$i_t = \sigma(\boldsymbol{W}_\text{i} x_t + \boldsymbol{U}_\text{i} h_{t-1} + b_\text{i}), \tag{7}$$

$$o_t = \sigma(\boldsymbol{W}_\text{o} x_t + \boldsymbol{U}_\text{o} h_{t-1} + b_\text{o}), \tag{8}$$

where, b_f, b_i and b_o gate of oblivion gate, input, output, gate bias; $\boldsymbol{W}_\text{f}$ and $\boldsymbol{U}_\text{f}$, $\boldsymbol{W}_\text{i}$ and $\boldsymbol{U}_\text{i}$, $\boldsymbol{W}_\text{o}$ and $\boldsymbol{U}_\text{o}$ gate of oblivion gate, input, output, respectively door weight matrix; σ is sigmoid activation function.

The calculation formula of the input cell state $\overline{\boldsymbol{C}}_t$ and the cell state $\overline{\boldsymbol{C}}_t$ at the current moment is:

$$\overline{\boldsymbol{C}}_t = \tanh(W_\text{c} x_t + U_\text{c} h_{t-1}), \tag{9}$$

$$\boldsymbol{C}_t = f_t C_{t-1} + i_t \overline{\boldsymbol{C}}_t, \tag{10}$$

where, tanh (·) is the activation function; $\boldsymbol{W}_\text{c}$, $\boldsymbol{U}_\text{c}$ are the weight matrix of the input state unit.

The output value h_t of the current network can be calculated by the following formula:

$$h_t = o_t \cdot \tanh(C_t). \tag{11}$$

LSTM neural network is trained by the time-based back propagation algorithm. Equations (6)-(11) are used to calculate the output of each unit, and then the error is calculated. The gradient is updated by back propagation, and the training process is iterated repeatedly until the error is less than the set threshold.

2.3 Model construction

Since the traffic flow data at signalized intersections have the characteristics of correlation, gradualism and periodicity, LSTM neural network can be used to forecast the traffic flow data as time series data. In order to improve the prediction accuracy, S-G filter is used to smooth the original data, and the smoothed data is more suitable for LSTM neural network to predict. The process of constructing the short-term traffic flow prediction model of SG-LSTM neural network is shown in Figure 2.

3 Multi-objective optimization of timing

3.1 The model of signal timing optimization is established

Average delay and average queuing times are cost-ori-

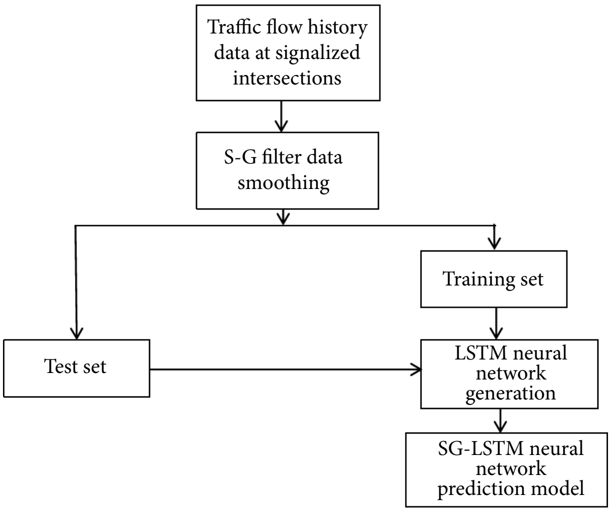

Fig. 2 SG-LSTM neural network prediction model

ented indexes, while traffic capacity is benefit oriented indexes. The smaller the cost-oriented indexes, the better, and the larger the benefit oriented indexes, the better. The above three indexes are used as timing optimization objectives to establish a multi-objective signal timing optimization model, as shown in Equations (12) and (13).

$$\min F(C, g) = \min[-Q(C, g), D(C, g), H(C, g)], \quad (12)$$

$$\text{s.t.} \begin{cases} g_{i,\min} \leqslant g_i \leqslant g_{i,\max} \\ \sum_{i=1}^{n} g_i + L = C \\ C_{\min} \leqslant C \leqslant C_{\max} \end{cases} \quad (13)$$

where, $Q(C, g)$ is the intersection capacity, pcu/h; $D(C, g)$ is the average delay of vehicles at the intersection, s; $H(C, g)$ is the average number of vehicle stops at the intersection; $g_{i,\min}$ is the minimum green time of phase i, s; $g_{i,\max}$ is the maximum green time of phase i, s; L is the lost time at the intersection, s.

3.2 The solution of the optimal timing model

Multi-objective optimization timing model is a nonlinear optimization problem, and there are many effective methods to solve this problem, such as ant colony algorithm, genetic algorithm, etc., these methods themselves have a certain complexity, while particle swarm optimization is easy to achieve without too many parameters, and is widely used in multi-objective optimization solution.

The particle swarm is first initialized as a group of random particles, and then seeks the optimal solution through continuous iteration. In each iteration, the particles update themselves by tracking individual extremum (p_{best}) and group extremum (g_{best}). The speed and position update formulas are shown in Equations (14) and (15) [12].

$$V_{id}^{k+1} = \delta V_{id}^{k} + c_1 r_1 (p_{\text{best}_{id}} - X_{id}^{k}) + c_2 r_2 (g_{\text{best}_{id}} - X_{id}^{k}), \quad (14)$$

$$X_{id}^{k+1} = X_{id}^{k} + V_{id}^{k}, \quad (15)$$

where, V_{id}^{k+1} for $k+1$ time iterative particle i first d flight velocity vector component; For the first X_{id}^{k+1} $k+1$ times iterative particle position vector of the first id component; δ inertia weight; c_1 and c_2 are two acceleration constants; r_1 and r_2 are two random numbers.

The flow of multi-objective particle swarm optimization is as follows:

Step 1: Initialize the particle swarm. Given the population size N, randomly generate the position X_i and velocity V_i of each particle.

Step 2: The fitness values of each particle Fitness [1, i], Fitness [2, i], Fitness [3, i], (i = [1, N]) were calculated by objective functions $-Q(C, g)$, $D(C, g)$ and $H(C, g)$ respectively.

Step 3: The individual extreme values $p_{\text{best}}[1, i]$, $p_{\text{best}}[2, i]$, $p_{\text{best}}[3, i]$ are obtained for each particle under the three objective functions.

Step 4: Three global extreme values $g_{\text{best}}[1]$, $g_{\text{best}}[2]$, $g_{\text{best}}[3]$ are obtained respectively for the three objective functions.

Step 5: The population extreme values g_{best} and distance $d_{g\text{best}}$ of the three objective functions are calculated.

Step 6: Calculating the distance between the extreme values of each individual particle $d_{p\text{best}[i]}$.

Step 7: Individual extremum $p_{\text{best}}[i]$ for calculating update speed V_i and position X_i.

Step 8: Update the position X_i and velocity V_i of each particle using the group extremum g_{best} and individual extremum $p_{\text{best}}[i]$.

Step 9: If the termination conditions are met, exit; otherwise, return to Step 2.

In Matlab software, particle swarm optimization algorithm is used to solve the signal timing optimization model. The program is mainly divided into three parts: objective function, particle swarm optimization algorithm, main function.

(1) Objective function: used to describe the model, define model variables and related calculation formulas. Input Equation (12), and take the minimum value of $F(C, g)$ as the objective function;The intersection green time g_i and signal cycle C were defined as variables in the population.

(2) Particle swarm optimization algorithm: compiling algorithm program, updating particle position and speed, and determining fitness function as $f_{it} = F(C, g)$.

(3) Main function: mainly used to store traffic flow, set the range of intersection green time g_i and signal cycle C in the model, and set the values of population number N, maximum number of iterations, acceleration constant c_1, c_2 and inertia weight δ.

4 Case analysis

4.1 Short-term traffic flow prediction

4.1.1 Data acquisition and preprocessing

This study utilized traffic flow data collected over a five-day period, from Monday to Friday, at the signalized intersection of Wangjing North Road and Guangze Road in Beijing, with observations recorded at 15 minute intervals. The channelization conditions of this intersection are shown in Table 1. The four phases are north-south straight, north-south left turn, east-west straight and east-west left turn respectively. The signal cycle is 157 s, and the effective green time of each phase is 57 s, 23 s, 42 s and 23 s. The traffic flow in the straight direction of the east-west entrance road at this intersection is selected as an example. There are 480 sets of data in total, 384 sets of data in the front are used as the training set, and 96 sets of data in the back are used as the test set.

The software origin uses S-G filter, a data smoothing method, to smooth the sample data. Figure 3 shows the sample data after smoothing.

Table 1 Road channelization conditions at signalized intersections

Inlet way	Crosswalk width (m)	Lane width (m)	Number of channelized lanes	Number of channelized lanes in each direction		
				Turn left	Go straight	Turn right
East import	36	3.5	4	1	2	1
Western import	36	3.5	4	1	2	1
South import	29	3.5	3	1	1	1
North import	29	3.5	3	1	1	1

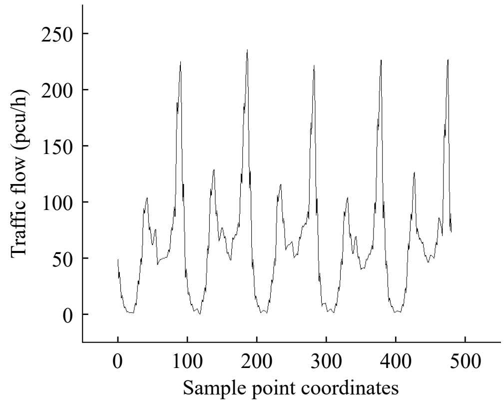

Fig. 3 Sample data after smoothing

4.1.2 Prediction result analysis

Matlab was used to set the first 80% of the data as the training set and the last 20% as the test set. The time step length was 4 and 1 step length was 15 min. In other words, the traffic flow data after smoothing in the past hour was used to predict the traffic flow in the next 15 min. Set the input dimension to 4, the output dimension to 1, and the number of iterations to 250. The SG-LSTM neural network can be used to obtain the comparison of the predicted value and the real value, as shown in Figure 4.

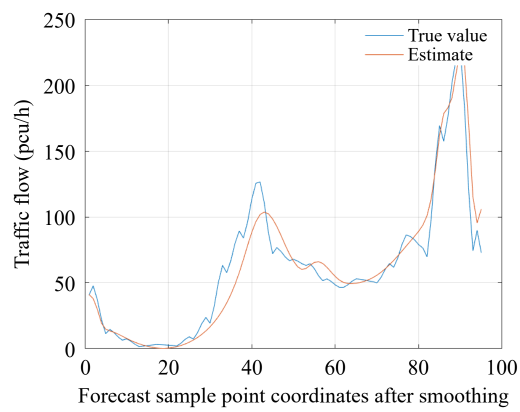

Fig. 4 Comparison between the predicted value and the real value of SG-LSTM neural network

The LSTM neural network is trained by using unsmoothed data and smoothed data of S-G filter respectively. The calculation results of performance indexes under the two conditions are shown in Table 2.

As can be seen from the above table, E_{RMS}, E_{MA} and E_{MAP} of the SG-LSTM neural network model are all better than those of the LSTM neural network model, and

Table 2 Performance specifications of SG-LSTM and LSTM neural network

Model	E_{RMS}	E_{MA}	E_{MAP} (%)
SG-LSTM neural network	8.0557	5.6008	4.1665
LSTM neural network	15.9864	8.6085	5.6534

the root mean square error of the SG-LSTM neural network model is almost doubled from 15.9864 to 8.0557, effectively improving the prediction accuracy. It can be seen that using S-G filter to smooth traffic flow data at signalized intersections can effectively reduce model prediction errors and improve model prediction accuracy.

4.2 Timing optimization of signalized intersection

As can be seen from the prediction results of SG-LSTM neural network, the traffic flow in the period from 22:00 to 22:15 is the largest. The prediction results of SG-LSTM neural network converted the flow of each inlet into the hourly traffic volume during this period are shown in Table 3.

Table 3 SG-LSTM neural network flow prediction data

Inlet way	Standard traffic flow (pcu/h)		Total
	Go straight	Turn left	
East import	824	324	1 148
Western import	288	92	380
South import	328	220	548
North import	204	152	356

Since the speed of vehicles entering the intersection is about 20 ~ 30 km/h, the capacity of a single lane is set at 1 450 pcu/h according to the theoretical capacity of one lane suggested in the Urban Road Design Code. Thus, it can be calculated that the maximum flow ratio of each phase is 0.226, 0.152, 0.284, 0.223 in turn, and the sum of the intersection flow ratio is 0.885. At this intersection, the traffic flow is in the peak period, and the total flow ratio and total saturation of the intersection are relatively high, both close to 0.9. Therefore, emphasis should be placed on improving the traffic capacity at this period. The weight of the capacity objective function should be greater than that of the other two objective functions.

For multi-objective optimization problems, it is very important to determine the weight relationship between various objective functions, and whether the weight relationship can be reasonably determined is the key to multi-objective optimization[13]. Fuzzy preference method was used to determine the weight of each objective function, and the weight coefficients of traffic capacity, average delay and average queue length were $\frac{1}{2}$, $\frac{1}{3}$ and $\frac{1}{6}$, respectively.

In this paper, the fuzzy compromise programming method is used for dimensionless processing of the three performance indicators, and the minimum solution of the distance between the three is taken as the optimal solution[14], transforming the multi-objective optimization problem into a single-objective optimization problem.

The short green time of each phase is calculated according to the Planning and Design Regulations for Urban Road Intersection:

$$g_{\min} = 7 + \frac{L_p}{V_p} - I \tag{16}$$

where, L_p is pedestrian crossing distance, m; V_p is pedestrian crossing speed, generally 1.2 m/s; I is the interval between green lights, s.

Write particle swarm optimization algorithm programming language in Matlab, set relevant constraint parameters and algorithm parameters. The minimum green time of the first and second phases is set to 28 s, the minimum green time of the third and fourth phases is set to 36 s, the maximum cycle time is set to 180 s, and the loss time L is set to 12 s. The particle size is set at 200, and each particle has four position components, respectively representing the effective green time of the first, second, third and fourth phases. The maximum number of iterations is 300, the acceleration factor c_1 and c_2 are both set at 1.5, and the inertia weight δ is set at 0.9.

The optimized signal timing parameters are as follows: the period C is 147 s, and the effective green time of the first phase to the fourth phase is 34 s, 28 s, 45 s and 28 s, respectively. The convergence curve of the objective function is shown in Figure 5.

4.3 Comparison of optimization results

In order to further verify the effectiveness of the proposed method, the signal timing before optimization is compared with the traditional Webster method. The signal timing parameters using Webster's method are as fol-

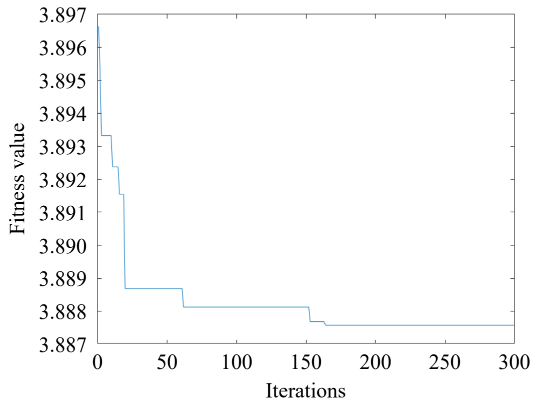

Fig. 5 Convergence curve of objective function

lows: the signal period C is 148 s, and the effective green time from the first phase to the fourth phase is 35 s, 23 s, 44 s and 34 s, respectively. As can be seen from the results in Table 4, after the particle swarm optimization algorithm is used for timing, the overall capacity of the intersection becomes 3 551, an increase of 9.1% compared with the current situation. The average delay was 47.85, down 18.6% from the current situation; The average number of stops was 0.81, down 8.9% from the current situation. At the same time, it can be seen that the optimization results of particle swarm optimization algorithm are superior to Webster's method in all aspects, which reflects the superiority of the algorithm.

Table 4 Comparison table of results of different timing schemes

Timing method	Effective green time (s)				Period (s)	Capacity	Average delay	Average number of stops
	First phase	Second phase	Third phase	Fourth phase				
Current situation	57	23	42	23	157	3 254	58.80	0.89
Webster	35	23	44	34	148	3 527	49.76	0.83
Particle swarm	34	28	45	28	147	3 551	47.85	0.81

5 Conclusions

In this paper, a LSTM neural network prediction method based on Savitzky-Golay filter is proposed, which preferentially processes traffic flow data, then predicts traffic flow, and compares its performance with that of the basic LSTM neural network. On the basis of the prediction results, a multi-objective optimal timing model was established. The fuzzy preference method was used to assign weights to the three objective functions. The fuzzy compromise programming method was used to carry out dimensionless processing on the three objective functions and transform them into single objective functions, which were solved by particle swarm optimization algorithm. The result of signal timing is compared with that of Webster signal timing and the original timing, and the following conclusions are drawn.

(1) It was found that RMSE, MAE and MAPE of SG-LSTM neural network were 8.055 7%, 5.600 8% and 4.166 5%, respectively, which were lower than that of basic LSTM neural network.

(2) The traffic capacity, average delay and average queuing times of the three performance indexes of the PSO signal timing results are 3 551, 47.85 and 0.81, respectively, which are superior to the Webster signal timing and the original time, proving the reliability of the algorithm and the constructed model.

Declaration of competing interest

The authors have no competing interests to declarethat are relevant to the content of this article.

References

[1] Zhong, C.H., Wang, M., Zhao, W., et al. Short-term traffic flow prediction of intersection based on LSTM[J]. Technology of Highway and Transport, 2019, 35(5): 119–123,131.

[2] Mou, L.T., Zhao, P.F., Xie, H.T., et al. T-LSTM: A long short-term memory neural network enhanced by temporal information for traffic flow prediction[J]. IEEE Access, 2019, 7: 98053–98060.

[3] Vlahogianni, E.I., Karlaftis, M.G., Golias, J.C. Optimized and meta-optimized neural networks for short-term traffic flow prediction: A genetic approach[J]. Transportation Research Part C: Emerging Technologies, 2005, 13(3): 211–234.

[4] Long, X.Q., Li, J., Chen, Y.R. Metro short-term traffic flow prediction with deep learning[J]. Control and Decision, 2019, 34(8): 1589–1600.

[5] Jing, B.B. Research on the fundamental theories and key

techniques of the bandwidth-based arterial signal coordination control[D]. Guangzhou: South China University of Technology, 2018.

[6] Chou, C.H., Teng, J.C. A fuzzy logic controller for traffic junction signals[J]. Information Sciences, 2002, 143(1/4): 73–97.

[7] Yang, X.W., Wang, Q.H., Xue, H.B., et al. A coordinated signal control method for arterial road of adjacent intersections based on the improved genetic algorithm[J]. Optik-International Journal for Light and Electron Optics, 2016, 127(16): 6625–6640.

[8] Cao, Y. Optimization of adaptive signal control using simulated annealing algorithm[J]. Journal of Transportation Engineering and Information, 2018, 16(1): 49–55,60.

[9] Dai, G.W. Research on short-term traffic flow forecast and signal control based on deep learning[D]. Lanzhou: Lanzhou Jiaotong University, 2021.

[10] Wang, C.X., Fu, S.N., Xiao, Z.P., et al. Long short-term memory neural network (LSTM-NN) enabled accurate optical signal-to-noise ratio (OSNR) monitoring[J]. Journal of Lightwave Technology, 2019, 37(16): 4140–4146.

[11] Qiu, X.P. Neural network and deep learning[J]. Journal of Chinese Information Processing, 2020(7): 1.

[12] Wen, Z. Proficient in MATLAB intelligent algorithm[M]. Beijing: Tsinghua University Press, 2015.

[13] Guan, Z.H., Kou, J.S., Li, M.Q. An evolutionary multiobjective optimization algorithm based on fuzzy preferences[J]. Journal of Tianjin University, 2002(3): 275–280.

[14] Xu, H.W., Huang, H.Z., Zhang, X. A fuzzy compromise programming approach to robust multi-objective optimization design[J]. Journal of Dalian University of Technology, 2007(3): 368–371.