

Modeling differential car-following behavior under normal and rainy conditions: A memory-based deep learning method with attention mechanism

Hai-Jian Bai(柏海舰), Chen-Chen Guo(过晨晨), Heng Ding(丁恒), Li-Yang Wei(卫立阳), Ting Sun(孙婷), and Xing-Yu Chen(陈星宇)

Citation: Chin. Phys. B, 2023, 32 (**6**): 060507. DOI: 10.1088/1674-1056/acaa2f
Journal homepage: http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

What follows is a list of articles you may be interested in

Bifurcation analysis of visual angle model with anticipated time and stabilizing driving behavior

Xueyi Guan(管学义), Rongjun Cheng(程荣军), and Hongxia Ge(葛红霞)
Chin. Phys. B, 2022, 31 (**7**): 070507. DOI: 10.1088/1674-1056/ac5606

Traffic flow prediction based on BILSTM model and data denoising scheme

Zhong-Yu Li(李中昱), Hong-Xia Ge(葛红霞), and Rong-Jun Cheng(程荣军)
Chin. Phys. B, 2022, 31 (**4**): 040502. DOI: 10.1088/1674-1056/ac3647

An extended smart driver model considering electronic throttle angle changes with memory

Congzhi Wu(武聪智), Hongxia Ge(葛红霞), and Rongjun Cheng(程荣军)
Chin. Phys. B, 2022, 31 (**1**): 010504. DOI: 10.1088/1674-1056/ac0eec

Heterogeneous traffic flow modeling with drivers' timid and aggressive characteristics

Cong Zhai(翟聪), Weitiao Wu(巫威眺), and Songwen Luo(罗淞文)
Chin. Phys. B, 2021, 30 (**10**): 100507. DOI: 10.1088/1674-1056/abf7ae

Stability analysis of traffic flow with extended CACC control models

Ya-Zhou Zheng(郑亚周), Rong-Jun Cheng(程荣军), Siu-Ming Lo(卢兆明), Hong-Xia Ge(葛红霞)
Chin. Phys. B, 2016, 25 (**6**): 060506. DOI: 10.1088/1674-1056/25/6/060506

Modeling differential car-following behavior under normal and rainy conditions: A memory-based deep learning method with attention mechanism

Hai-Jian Bai(柏海舰)[1,2,†], Chen-Chen Guo(过晨晨)[1], Heng Ding(丁恒)[1], Li-Yang Wei(卫立阳)[1,2], Ting Sun(孙婷)[1], and Xing-Yu Chen(陈星宇)[1]

[1] *School of Automotive and Transportation Engineering, Hefei University of Technology, Hefei* 230009, *China*
[2] *State Key Laboratory of Cognitive Intelligence, Hefei* 230009, *China*

(Received 20 September 2022; revised manuscript received 7 December 2022; accepted manuscript online 9 December 2022)

In order to analyze and learn the difference in car-following behavior between normal and rainy days, we first collect car-following trajectory data of an urban elevated road on normal and rainy days by microwave radar and analyze the differences in speed, relative speed, acceleration, space headway, and time headway among data through statistics. Secondly, owing to the time-series characteristics of car-following data, we use the long short-term memory (LSTM) neural network optimized by attention mechanism (AM) and sparrow search algorithm (SSA) to learn the different car-following behaviors under different weather conditions and build corresponding models (ASL-Normal, ASL-Rain, where ASL stands for AM-SSA-LSTM), respectively. Finally, the simulation test shows that the mean square error (MSE) and reciprocal of time-to-collision (RTTC) of the ASL model are better than those of LSTM and intelligent diver model (IDM), which is closer to the real data. The ASL model can better learn different driving behaviors on normal and rainy days. However, it has a higher sensitivity to weather conditions from cross test on normal and rainy data-sets which need classification training or sample diversification processing. In the car-following platoon simulation, the stability performances of two models are excellent, which can describe the basic characteristics of traffic flow on normal and rainy days. Comparing with ASL-Rain model, the convergence time of ASL-Normal is shorter, reflecting that cautious driving behavior on rainy days will reduce traffic efficiency to a certain extent. However, ASL-Normal model produces a more severe and frequent traffic oscillation within a shorter period because of aggressive driving behavior on normal days.

Keywords: car-following behavior, deep learning, weather condition, heterogeneous traffic flow

PACS: 05.60.–k, 45.70.Vn **DOI:** 10.1088/1674-1056/acaa2f

1. Introduction

The influence of rainy conditions on drivers is significant. Raining will reduce the driver's perception of surrounding traffic conditions and the adhesion coefficient between tires and roads, which increase the braking distance and the driving safety risk. In such an adverse weather, drivers tend to adopt a more conservative driving behavior.[1–4] They will reduce the risk of driving by reducing the speed and increasing the space headway.[5] This adjustment of micro driving behavior will lead to significant changes in macro traffic flow characteristics on rainy days, such as traffic efficiency reduction and traffic oscillation.[6] Therefore, modelling car-following behavior on rainy days can more intuitively show the interaction relationship between vehicles on rainy days, which is significant in analyzing rainy driving behavior and simulating rainy traffic. At the same time, it can also improve the rainy driving tips for automatic driving.

According to different research methods, car-following models have two categories. One is the traditional principle-driven car-following model established by mathematical and physical methods, such as the classical IDM model,[7] full velocity difference (FVD) model,[8] Gazis–Herman–Rothery (GHR) model,[9] generalized force model (GFM),[10] optimal velocity model (OVM),[11] Gipps model,[12] and Newell model.[13] Most of researches into adverse weather focused on optimizing the parameters of the traditional car-following model to adapt to the weather. For example, Soria *et al.*[3] and Hammit *et al.*[4] used natural driving data to calibrate and optimize the parameters of the Gipps model, which better reflected the relatively conservative driving behavior of drivers on rainy days, and also verified the heterogeneity of car-following behavior in adverse weather conditions. Huang *et al.*[14] used the driving simulator to collect the car-following data at different fog levels and analyzed the drivers' heterogeneous driving behavior on foggy days. They improved IDM model by adding an unreasonable speed adjustment item with a small probability and setting the time-varying desired time headway at different foggy levels. The new model can better describe the car-following scene in foggy conditions.

These studies show that the heterogeneous car-following behaviors under adverse weather conditions are significant, which needs to be modeled separately. Owing to the constraints of theoretical formula, the principle-driven model is

[†]Corresponding author. E-mail: baihaijian@hfut.edu.cn

© 2023 Chinese Physical Society and IOP Publishing Ltd

not comprehensive in describing the car-following behavior under adverse weather conditions: it lacks some characteristics of human drivers. In addition, when using this kind of model to simulate vehicle microscopic trajectory, each trajectory needs to carry out complex parameter calibration separately in order to obtain better accuracy, which makes the use of the model very cumbersome.

In recent years, with the development of artificial intelligence and big data technology, another kind of data-driven nonparametric car-following model has developed rapidly . Data-driven models can be divided into support vector regression models,[15] instance learning models (including local regression and K-nearest neighbor regression),[16–18] probability estimation models (including Bayesian filtering and Kalman filtering),[19,20] neural network models,[21–24] and deep learning models (in-cluding reinforcement learning).[25–34] The support vector regression is a small sample learning method with solid theoretical basis, which is difficult to learn large sample data or multi-dimensional data. The instance learning is a simple method that average the data similar to the current driving scene, but it may not be accurate for using average driving behavior to simulate a single trajectory. The probability estimation method is only a simple fitting for the driving behavior of a single trajectory. Compared with other date-driven models, the neural network model can carry out high-precision learning for large sample and multi-dimensional data, which is more widely used.

The earliest neural network used for car-following modelling is an artificial neural network (ANN).[21–24] Kehtarnavaz *et al.*[21] creatively developed a car-following model with ANN. Panwei and Dia[22] proved that the car-following model based on the ANN is superior to the traditional Gipps model in terms of accuracy and robustness. However, the structure of ANN is simple, which can only carry out shallow learning and cannot consider the historical time-series characteristics of data.

With the emergence of new deep learning algorithms, more car-following models have been developed and applied to traffic simulation, such as recurrent neural network (RNN),[25] LSTM,[26–28] gated recurrent unit (GRU),[29–31] sequence to sequence (Seq2Seq) learning,[32] and reinforcement learning (RL)[33,34] car-following models. Zhou *et al.*[25] established the car-following model with RNN and proved that the RNN model is equivalent to the classical IDM and OVM model in predicting the trajectory accuracy but better in predicting different types of traffic oscillations caused by different driving behaviors. The RNN can carry out deep learning and process time-series data, but it is prone to gradient disappearance or gradient explosion. The LSTM is a particular type of RNN, which can solve the problem of gradient explosion or gradient disappearance. Huang *et al.*[27] built a car-following model by using LSTM-NN. From the perspective of human asymmetric driving behavior, they found that the LSTM model is superior to the RNN model and AFVD (asymmetric full velocity difference) model in reproducing human micro driving behavior. The GRU is a simplified variant of the LSTM, Wang *et al.*[29] established a car-following model by using GRU-NN and took the following vehicle speed, relative speed and space headway in recent time intervals as the model's input to simulate the driver's memory effect. The training and calculation speed of GRU are better than those of LSTM, but the LSTM has stronger learning capability and higher model accuracy. Considering the multi-step decision-making process in the process of human driving, Ma and Qu[32] proposed a car-following model based on Seq2Seq learning. The model considered memory effect and reaction delay, which can make a multi-step prediction. But the encoding and decoding process of Seq2Seq may cause the loss of data information. RL is a special deep learning method, Zhu *et al.*[33] took the distribution of jerk, TTC and time headway as reward-punishment function, and used RL to establish a car-following model. The model has better safety, comfort, and efficiency by comparing the human natural driving data. But it is still unknown whether the construction of car-following environment and the design of reward-punishment function are reasonable.

Through directly learning the inherent law of the car-following data on rainy days, these deep learning model can effectively reflect the characteristics of human drivers, which have been observed from real data, such as the asymmetry of acceleration and deceleration,[27] driver's memory effect,[29] and reaction delay.[32] The model has a high accuracy, which can better reproduce the real car-following behavior on rainy days. Meanwhile, using these model does not need to consider complex formulas nor parameter calibration, and the car-following behavior can be extracted and learned from observation data at any time. Compared with the traditional principle-driven model under adverse weather conditions, the deep learning model shows higher anthropomorphic accuracy, better generalization capability, stronger flexibility, and practicability. However, these models are still in the exploratory stage. Most relevant researches are based on the existing open-source or experimental dataset to verify the advantages of artificial intelligence algorithms in car-following behavior decision-making. Owing to the lack of high-precision car-following data, there is less research on whether artificial intelligence algorithms can learn heterogeneous driving behavior on rainy days.

Therefore, considering the data collection difficulty of the vehicle on rainy days, this work uses microwave radar to collect vehicle trajectory data on an elevated road in China and build training and testing data-sets for normal and rainy days, respectively. And then, the different characteristics of traffic

flow in different weather conditions are verified through statistical analysis. Form the comparison among various data-driven models, the LSTM is finally used for car-following modeling. Owing to the fact that the accuracy of initial neural network can be improved, the AM and SSA are used to optimize the initial LSTM and successfully improve the model' accuracy. Based on the data-sets, two ASL models (ASL-Normal, ASL-Rain) are trained respectively to learn real car-following driving on normal and rainy days. The sensitivity of the ASL model to weather conditions is further verified by cross-testing with normal and rainy data-sets. By comparing with the IDM model and LSTM model, the performance of the ASL model in trajectory prediction on normal and rainy day's car-following scenarios is conformed. Finally, the real traffic scenes of normal and rainy days are reproduced through the disturbance platoon simulation to verify the stability and reliability.

2. Data preparation

2.1. Data collection

At present, the main methods to obtain the car-following trajectory data are as follows. (i) Set up a camera at a high altitude to take videos of the road section, and then extract trajectory by using image recognition and processing software, such as NGSIM data-set.[25–32] (ii) Use the unmanned aerial vehicle (UAV) aerial photography to obtain the car-following data of the road section, such as the highD data-set.[35] (iii) Through the driving simulator, hire drivers to conduct driving experiments to collect car-following data.[36] (iv) Install radar and sensors on a test vehicle to obtain the car-following data of the vehicle.[37] (v) Use microwave radar to obtain the car-following data of the road section.[38] For methods (i) and (ii), video image acquisition is difficult due to the low visibility on rainy days, and the error of trajectory extraction is significant. For methods (iii) and (iv), the sampling scale of the simulator and test vehicle is limited, and the driving behavior is easy to be disturbed by external factors. Therefore, this work selects microwave radar to obtain the natural car-following data through the sampling, testing, and comparing the trajectory on rainy days.

The effective detection range of millimeter microwave radar is 200 m, and the data acquisition frequency is 50 Hz. The radar is installed on the traffic monitoring pole of North–South 1st Elevated Road in Hefei, Anhui Province, China. The road conditions of the elevated road section are normal, and there are ten lanes in the double direction as shown in Fig. 1.

The four lanes close to the road centerline in the north direction are selected as the data collection lanes, and we pick out two hours of data from 5:00 to 6:00 on July 27th and July 30th, 2021. July 27th was a moderate to heavy rain with 9.4 -mm–15.4 mm of precipitation per 12 hour from 5:00 to 6:00, while July 30th was sunny. The rainy data include the trajectory data of 4672 cars, and the sunny data includes the trajectory data of 5078 cars.

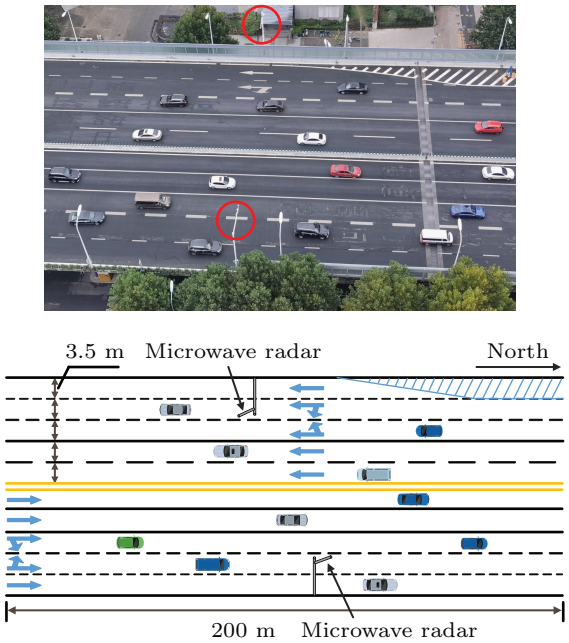


Fig. 1. Microwave radar installation section.

2.2. Data preprocessing

The data directly collected by the radar includes two behaviors of car-following and lane-changing. It is necessary to extract the car-following trajectory from the raw data. The specific extraction principles are as follows. (I) The car-following time of extracted trajectory is greater than or equal to 10 s. If the time is too short, the sampling points will be insufficient. (II) The time headway does not exceed 5 s, otherwise, there will be no obvious following characteristics.[39] (III) The vehicle's acceleration shall be distributed between -4 m/s^2 and 4 m/s^2.[40] (IV) The frequency of data is 10 Hz. It has proved that using the data in steps of 0.1 s to train the neural network can improve the learning ability of the network.[24]

Based on these principles, 800 trajectories were extracted on normal and rainy days, respectively. However, the raw data are not smooth, and the direct use of raw data in model training will reduce the learning accuracy.[41] Therefore, this work uses symmetric exponential moving average (sEMA) filter for data smoothing.[42] Equations (1) and (2) show the specific formula. Figure 2 shows the smoothing results of vehicle acceleration and speed.

$$\bar{x}_a(t_i) = \frac{1}{Z}\sum_{i-D}^{i+D} x_a(t_{i-D})\, \mathrm{e}^{-D\cdot\Delta t/T}, \tag{1}$$

$$Z = \sum_{i-D}^{i+D} \mathrm{e}^{-D\cdot\Delta t/T}, \tag{2}$$

where $\bar{x}_a(t_i)$ is the smoothed data of vehicle a at time t_i; $x_a(t_{i-D})$ is the raw data of vehicle a at time t_{i-D}; Δt is the time step and set to 0.1 s in this works; D is the smoothing kerneland set to 10 in the work; T is the smoothing time width.

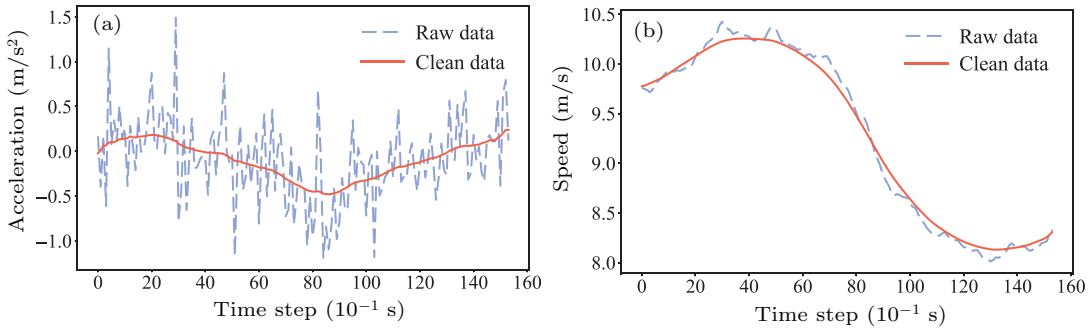


Fig. 2. Comparison between raw and smoothed data (vehicle 1544 on normal day): (a) acceleration smoothing, (b) speed smoothing.

2.3. Characteristics of car-following data

In this work, the five characteristics of vehicle speed, acceleration, relative speed, space headway, and time head-way are analyzed statistically to verify the difference in driver's behavior on normal and rainy days. It can be seen from Table 1 and Fig. 3(a) that compared with sunny drivers, the rainy drivers tend to drive at a lower speed. The mean, maximum, minimum, and median speeds are lower than those under normal conditions, and the average speed decreases by 12.1%. there have been researches showing that the free flow speed under rainy conditions decrease by 3.7%–11.0%.[1,3,4] Some studies also show that rainy conditions will reduce the free-flow speed of expressway by 3.0%–16.2%,[43] and urban roads by 4.0%–17.0%.[44]

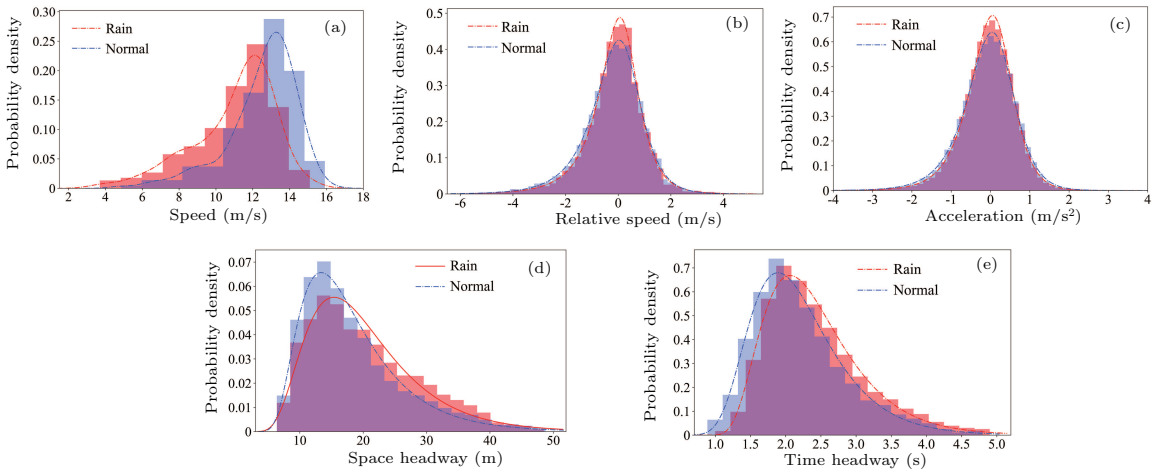


Fig. 3. Statistical distribution of characteristic variables of vehicle trajectory on normal and rainy days. (a) Speed distribution, (b) relative speed distribution, (c) acceleration distribution, (d) space headway distribution, (e) time headway distribution.

Table 1. Statistic results of characteristic variables of vehicle trajectory on normal and rainy days.

Variable	Weather	Mean	SD	Max	Min	Median	Mean difference
Speed v (m/s)	Normal	12.4960	1.8863	15.9130	4.8636	12.9423	–12.09%
	Rain	10.9851	2.2178	15.1000	3.6964	11.5865	
Relative speed Δv (m/s)	Normal	–0.2128	1.1798	4.8453	–5.9734	–0.1094	–
	Rain	–0.1427	1.1138	4.4095	–5.9558	–0.0428	
Acceleration a (m/s^2)	Normal	–0.0836	0.7380	3.8536	–3.9991	–0.0143	–
	Rain	–0.0772	0.6811	3.6280	–3.8416	–0.0328	
Space headway Δx (m)	Normal	18.4299	8.1297	48.7961	6.1154	16.2580	10.00%
	Rain	20.2725	8.5074	48.5074	6.3725	18.6023	
Time headway Δt (s)	Normal	2.2044	0.6781	4.7994	0.8870	2.0773	9.96%
	Rain	2.4239	0.6992	4.8778	0.9971	2.2815	

At the same time, the average relative speed on rainy days is lower than that under the normal conditions, and the following vehicle tends to maintain a low-speed difference from the lead vehicle to avoid frequent acceleration and deceleration. The driving behavior on rainy days tends to be conservative. The standard deviation of acceleration under rainy conditions is smaller than under normal conditions, and its acceleration distribution is more concentrated near the mean value, which is in line with drivers' more conservative driving behavior.

In order to compare the difference in space headway be-

tween normal days and rainy days, this work selects the data with similar speeds in the car-following trajectory to analyze their space headway and time headway, as shown in Figs. 3(d) and 3(e). At similar speeds, drivers tend to maintain a larger space headway and time headway on rainy days, about 10% lower than those on normal days. The influence of rainy conditions on driving behavior is significant.

3. AM-SSA-LSTM-NN model

Car-following data are the typical data with time-series characteristics. Compared with other deep learning models, LSTM model has strong learning capability and high prediction accuracy to time-series data. It can more accurately describe the car-following behavior and the model itself has no obvious shortcomings. Moreover, the LSTM with the forget gate mechanism can theoretically consider the historical data of more than 1000 time steps. It can learn historical information and delete outdated information, which is similar to human's memory driving behavior. Relevant papers have proved that the car-following model considering the driver's memory effect can better reproduce the real state of traffic flow.[29] Therefore, we choose the optimized LSTM for car-following modeling.

3.1. LSTM neural network structure

A complete LSTM neural network usually includes an input layer, one or more hidden layers and an output layer. The input layer and output layer control the input and output of the model, and the hidden layer controls the data generalization feature learning. The hidden layer is an essential part of the neural network structure, which is connected by internal memory block units. Each memory block unit includes the input gate, output gate, forgetting gate and memory cell unit.[27] Figure 4 shows the specific structure of the unit.

In Fig. 4, x_t is the vector input at time t, c_{t-1} and a_{t-1} are the output vectors of the previous time $t-1$, f_t, i_t, c_{t0}, and o_t are all an internal control variable, c_t and a_t are the current output vectors, σ and tanh are sigmoid activation function and hyperbolic tangent activation function, respectively, which are to convert the inputs into the value of (0, 1) and $[-1, 1]$ interval. The σ and tan are expressed as

$$\sigma(x) = \frac{1}{1+\mathrm{e}^{-x}}, \tag{3}$$

$$\tanh(x) = \frac{\mathrm{e}^{x}-\mathrm{e}^{-x}}{\mathrm{e}^{x}+\mathrm{e}^{-x}}. \tag{4}$$

The specific calculation process of the LSTM neural network is as follows:

(i) forget gate

$$f_t = \sigma(W_{\mathrm{fx}}x_t) + W_{\mathrm{fa}}a_{t-1} + W_{\mathrm{fc}}c_{t-1} + b_{\mathrm{f}}; \tag{5}$$

(ii) input gate

$$i_t = \sigma(W_{\mathrm{ix}}x_t) + W_{\mathrm{ia}}a_{t-1} + W_{\mathrm{ic}}c_{t-1} + b_{\mathrm{i}}, \tag{6}$$

$$c_{t0} = \tanh(W_{\mathrm{cx}}x_t + W_{\mathrm{ca}}a_{t-1} + b_{\mathrm{c}}); \tag{7}$$

(iii) memory cell unit

$$c_t = f_t \otimes c_{t-1} + i_t \otimes c_{t0}; \tag{8}$$

(iv) output gate

$$o_t = \sigma(W_{\mathrm{ox}}x_t) + W_{\mathrm{oa}}a_{t-1} + W_{\mathrm{oc}}c_{t-1} + b_{\mathrm{o}}, \tag{9}$$

$$a_t = W_{\mathrm{ao}}o_t + b_{\mathrm{a}}, \tag{10}$$

where W_{fx}, W_{fa}, W_{fc}, W_{ix}, W_{ia}, W_{ic}, W_{cx}, W_{ca}, W_{ox}, W_{oa}, W_{oc}, and W_{ao} represent the weight matrix corresponding to each layer, b_{f} , b_i, b_{c}, b_{o}, and b_{a} are the offset weights corresponding to each gate.

The LSTM neural network has many-to-many and many-to-one prediction structures according to different data input methods. Many-to-many structure is a structure in which each sequence input corresponds to the output of a sequence, while many-to-one structure is a structure in which multiple sequence inputs correspond to the output of a sequence.

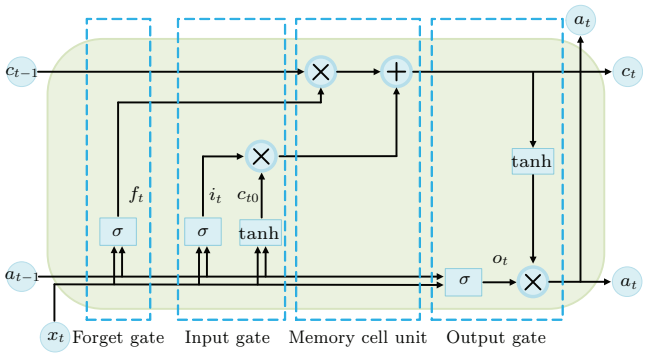


Fig. 4. LSTM memory block cell structure.

In this paper, the many-to-one prediction structure is adopted, and one output is obtained for each sample data input. The sample data are obtained by using the sliding window method, and the window size is 40 and the sliding step is 1. Each sample contains 40 time steps of relative speed, space headway and following vehicle speed data, and 57200 samples are obtained on normal and rainy days respectively. The reason for selecting 40 time step data as a sample is that relevant papers have proved that the driver's driving behavior is affected by the previous driving memory,[29] and the input of 40 time steps data exactly corresponds to the driver's driving memory before 4 s.

3.2. Input and output structure

In the actual traffic scene, car-following behavior arises from the interaction between the lead vehicle and the following vehicle. Most of the existing car-following models describe this interaction as the stimulation of the lead vehicle to the following vehicle and the response of the following vehicle

to the lead vehicle,[7–11] and the related neural network models usually use the relative speed, the space headway and the speed of the following vehicle to define this stimulation and use the speed change or acceleration change of the following vehicle to define this response.[27–30,32] So, this work also selects the relative speed, space headway, and following vehicle speed as the model's input, and the acceleration of the following vehicle as the model's output. Figure 5 shows the ASL model's training prediction framework, many-to-one structure and specific input and output dimensions.

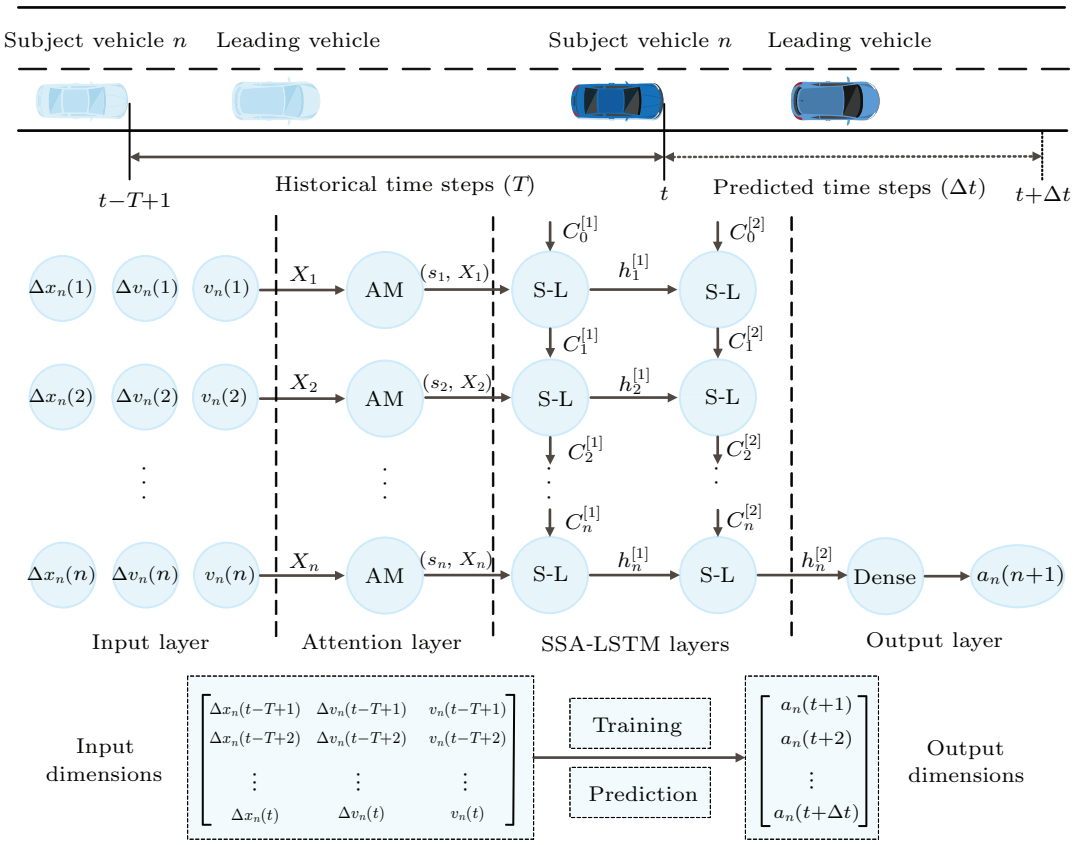


Fig. 5. ASL training and prediction framework.

3.3. Sparrow search algorithm

The sparrow search algorithm is a swarm intelligence optimization algorithm inspired by the foraging behavior and anti-predatory behavior of sparrow.[45] It divides the sparrows into three categories: explorers, participants, and vigilantes. The explorers have a high energy reserve, they are responsible for searching food rich areas and providing search areas and directions for the participants. They can transform with the participants, but their proportion in the population remains the same. The vigilantes are responsible for telling other companions whether there are predators around. If the alarm value is greater than the set value, the sparrows will quickly move to the safe area. This unique exploration–participation guidance mechanism makes the SSA's convergence speed and accuracy better than others. The positions update of sparrows after they have been aware of the danger reduces the probability of the algorithm falling into local optimization.

In this work, we use the SSA to optimize the LSTM neural network with the goal of finding a set of optimal hyperparameters to minimize the error of the neural network. First, we randomly select three quarters of the sample data as the training set and use the remaining sample data as the testing set to identify the model's performance. Then, we use different positions of sparrows to represent different hyperparameter combinations and select the mean square error (MSE) function of the testing set as the overall fitness function. The lower the fitness function value, the better the optimization effect is. Finally, we obtain the optimal hyperparameter combination through continuous iteration, and use these hyperparameters to build the optimal model. Table 2 shows the parameter settings of the SSA, where Itr represents the maximum iterations; Pop denotes the population of sparrows; Dim is the search dimensions; Pe, Pv, and Rv are referred to as the proportion of explorers, the proportion of vigilantes, and the alarm value, respectively. Figure 6 shows the implementation process of the SSA-LSTM.

Table 2. Parameters setting of the SSA.

Itr	Pop	Dim	Pe	Pv	Rv
50	20	8	20%	10%	0.8

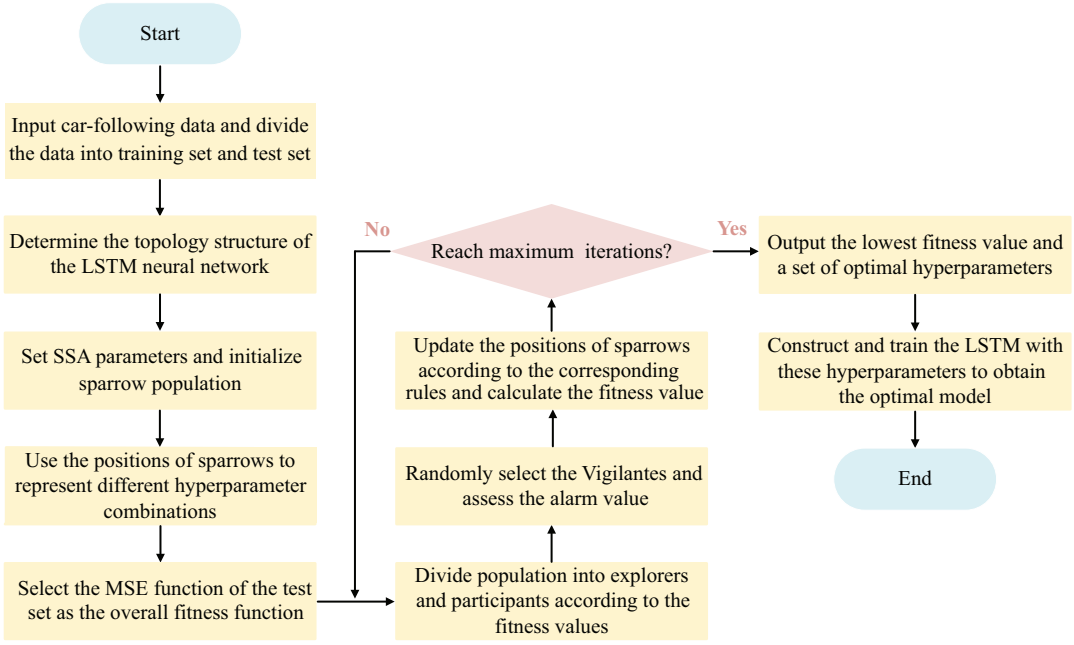


Fig. 6. Implementation process of SSA-LSTM.

3.4. Attention mechanism

The attention mechanism was first applied to natural language translation[46] to calculate the weight of each feature when the neural network updates the state each time and feeds it back to tell the network the different importance of the input data. The simplified calculation principle is shown in Eqs. (11) and (12). Combining the attention mechanism with LSTM neural network can enhance the feature learning capability of the network for the input data and make the performance of the model better. At present, this method is applied to the field of transportation and proves to perform well in vehicle speed prediction, traffic flow prediction, and lane change intention recognition. In this work, the attention mechanism is added before the SSA-LSTM layer. Figures 7 and 8 show the specific implementation process and calculation results.

$$a_i = \mathrm{softmax}(e_i e_m) = \frac{\exp(e_i^{\mathrm{T}} e_m)}{\sum_{i=1}^{t} \exp(e_i^{\mathrm{T}} e_m)}, \tag{11}$$

$$s = \sum_{i=1}^{t} \alpha_i X_i, \tag{12}$$

where α_i is the normalized attention coefficient array; e_i is the hidden state output calculated by the LSTM layer; e_m is the randomly initialized attention weight array; s is the output of the attention layer; X_i is the input array.

As shown in Fig. 7, the initial input data format of the input layer is an array, and the shape refers to batch_size, time_step, and input_dimension. First, using the permute layer to convert the third and the second dimension into the shape (batch_size, input_dimension, time_step). After the input data format is finished in the reshape layer, the feature weight for the dense (Softmax) layer is calculated by feeding the input data into the activation function. Then the permute layer is used to restore the calculated weight data shape to the initial input data shape, and the obtained result together with the input data is substituted into the multiply layer to obtain the final attention weights. Next, the attention weights and input data are substituted into SSA-LSTM layer. In the following training, the network will pay more attention to these feature data with large attention weights, which will improve the learning effect of the network.

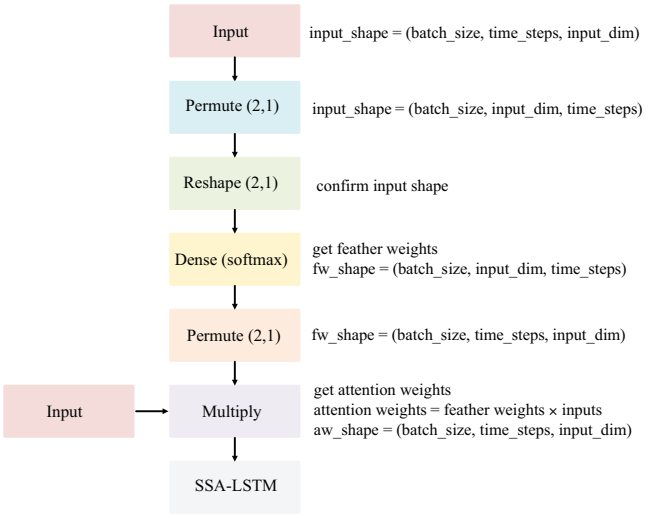


Fig. 7. Implementation process of attention mechanism.

Samples N1 and N2 are randomly selected from training data samples, and it can be seen from Fig. 8(a) that the attention weights of Δx, Δv, and v are different in different samples. It can be seen from Fig. 8(b) that the attention weights of data corresponding to different time steps in a sample are different, and the change trends of the attention weight curves corresponding to the space headway Δx, relative speed Δv, and fol-

lowing vehicle speed v are different. The difference in attention weight ratio among three input features in the spatial dimension and the difference in attention weight ratio among input features in the time step dimension indicate that the attention mechanism plays a good role. The attention mechanism can be used to successfully calculate the attention weights of the time step dimension and the input spatial dimension, and the SSA-LSTM model will perform more targeted learning according to these weights.

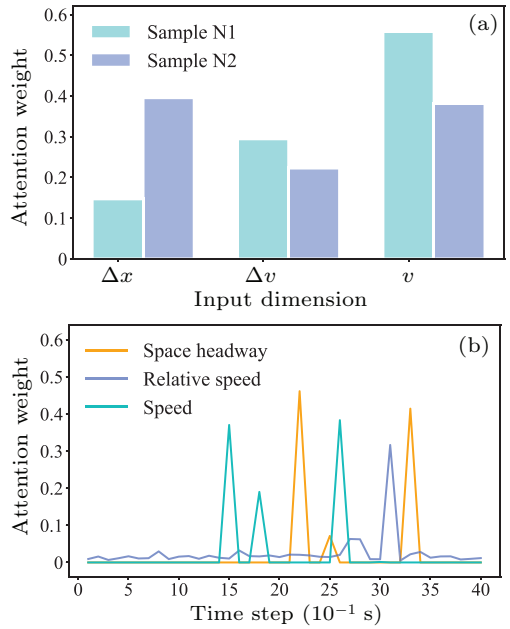


Fig. 8. Calculation results of attention weight in different dimensions. (a) Input spatial dimension, (b) time step dimension.

3.5. Hyperparameter optimization result and analysis

The LSTM has many parameters and configurations, such as activation function, loss function, optimization algorithm, number of hidden layers, number of neural units in each layer, learning rate, batch-size and epoch. These parameters and configurations can be optimized to make the neural network's performance better.

(I) Activation function

Activation function is used to complete the data's nonlinear transformation. The commonly used activation functions in neural networks include sigmoid, tanh, Relu, and Leaky Relu. These functions have different performances for different data. Generally, the default sigmoid function and tanh function can be used.

(II) Loss function

Loss function is an error function used to measure the difference between the predicted value of the neural network and the actual value. Usually, the MSE is selected as loss function, and its specific formula is shown below:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(a_i^{\mathrm{sim}} - a_i^{\mathrm{obs}}\right)^2, \tag{13}$$

where a_i^{sim} is the acceleration prediction of the model, a_i^{obs} is the acceleration of the raw data, N is the number of samples in each time step.

(III) Optimization algorithm and learning rate

Optimization algorithm of the neural network is to find the minimum value of the loss function. We use the adaptive moment estimation (Adam) algorithm because it has the best robustness for each parameter. The learning rate is the step size of each iteration of the optimization algorithm. The appropriate learning rate will make the loss function decrease rapidly with time going by. If the learning rate is too small, the algorithm may need more iterations to find the minimum value. If it is too large, the algorithm may skip the minimum value and oscillate around the minimum value. Generally, the learning rate is set to [0.001–0.1].

(IV) Batch-size and epoch

In neural network, batch-size represents the amount of data inputted into the network for each batch of training, and its value affects the efficiency of model training. According to the amount of car-following data input, set it to 32–128] in this network. Epoch represents the generation of model training. In order to avoid overfitting, in this work the early-stopping mechanism of the neural network is used. When the loss function value of the testing set does not continue to decline for more than 3 epochs, the network will automatically stop training.

(V) Dropout rate

The dropout mechanism of the neural networks can suppress the serious overfitting by randomly discarding a certain proportion of hidden neuron outputs in the training process. Too small a dropout rate may not be enough to suppress the overfitting, while too large a dropout rate will reduce the learning capability of the network and reduce the accuracy of the model. Dropout rate is generally set to 0.1–0.5.

(VI) Number of hidden layers and number of neural units in each layer

The number of hidden layers of the neural network and the number of neural units in each layer determine the learning capability of the model. Adding more hidden layers and increasing the number of neural units in each layer will increase the model's learning capability, but it will lead to serious overfitting . Therefore, it is necessary to determine the optimal collocation according to the actual situation. Referring to relevant literature,[20,22] adding one to three layers to the hidden layer is suitable, and the number of neural units can be 2–128.

In this work, the SSA is used to find the optimal combination of learning rate, batch-size, dropout rate, and the number of neurons in the three hidden layers. The overall training process of AM-SSA-LSTM model is as follows:

(i) Randomly scramble the data-set, divide the training set and testing set according to the ratio of 3:1, and record the scrambled data.

(ii) Build the overall network structure, add a dropout layer behind each LSTM layer, conduct pre training and record the weight of the first epoch.

(iii) Take the scrambled training set data as the input, and call back the weight of the first epoch as the initial weight.

(iv) Add the early-stopping mechanism to monitor the loss function of testing set, set up a random number seed to make the subsequent random numbers consistent, and use the SSA to optimize the hyperparameters of the model.

(v) After getting the optimal hyperparameters combination, input it into the constructed neural network structure and add attention mechanism to the neural network.

(vi) Establish a cycle to continuously train the network. When the loss function values of the training set and the testing set are close and maintain at a very low level, stop training, and save the optimal model.

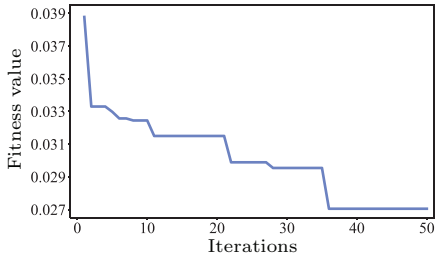


Fig. 9. Fitness function iterative results of SSA.

As shown in Fig. 9, the SSA successfully reduces the fitness value to a very low level after being iterated continuously. It is indicated that the algorithm has found the optimal combination of hyperparameters. To further prove the influence of the SSA and AM on LSTM, we select the MSE, mean absolute error (MAE), mean absolute percentage error (MAPE), and theil inequality coefficient (TIC) to compare the accuracy of LSTM, SSA-LSTM, AM-LSTM, and AM-SSA-LSTM. The lower the value of each index, the higher the accuracy of the model is. The formula of each index is as follows:

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|a_i^{\mathrm{sim}} - a_i^{\mathrm{obs}}|, \tag{14}$$

$$\mathrm{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{a_i^{\mathrm{sim}} - a_i^{\mathrm{obs}}}{a_i^{\mathrm{obs}}}\right|, \tag{15}$$

$$\mathrm{TIC} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(a_i^{\mathrm{sim}} - a_i^{\mathrm{obs}}\right)^2}}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(a_i^{\mathrm{sim}}\right)^2} + \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(a_i^{\mathrm{obs}}\right)^2}}. \tag{16}$$

Table 3 shows the calculation results of each index, in which each value is the average value of the testing set calculation results after being trained ten times. Comparing with the initial LSTM, each index values of the SSA-LSTM and AM-LSTM are low, indicating that both the SSA and AM have successfully optimized the LSTM. The AM-SSA-LSTM has the lowest index value, indicating that using the SSA and AM together to optimize the LSTM can further improve the accuracy of the model.

Table 3. Calculation results of different indexes.

Model	MSE	MAE	MAPE	TIC
LSTM	0.0305	0.1006	1.4230	0.2339
SSA-LSTM	0.0292	0.0967	1.3316	0.2282
AM-LSTM	0.0282	0.0949	1.3227	0.2261
A-S-L	0.0274	0.0920	1.3098	0.2223

Table 4. Parameter optimization results of ASL car-following model.

Parameter	Value	Parameter	Value
Activation function	sigmoid tanh	learning rate	0.0021
Loss function	MSE	time step	0.1 s
Optimization algorithm	Adam	epoch	36
Number of hidden layers	3	batch-size	108
number of neural units	126, 107, 83	dropout	0.1329, 0.1065, 0.3764

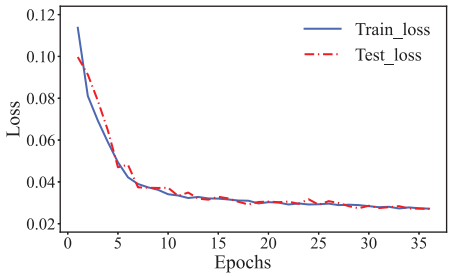


Fig. 10. ASL model training and testing error.

Figure 10 shows the variation of ASL model's loss function of training and testing with epoch. After 36 epochs, the loss function values of training and testing set are very low and maintained at a similar level, indicating that no overfitting problem has occurred, and the ASL model has successfully learned and mastered the rules of the input data. The car-following data on normal days and on rainy days finally adopt the same ASL network structure after being tested and optimized continuously, and Table 4 shows the optimization results of various parameters.

4. Simulation test and analysis

4.1. Cross test on normal and rain data-sets

Single vehicle simulation steps of the many-to-one ASL model is as follows. (i) Input the relative speed, space headway and following vehicle speed of 40-time steps, then the

model predicts the acceleration of the next time step. We can calculate the speed, distance, relative speed, and space headway of the next time step from this acceleration. (ii) Eliminate the data of the first-time step and add new data, then input the fused data of 40-time steps into the model to calculate the following vehicle acceleration of the next time step. (iii) Continuous iteration can simulate complete following vehicle trajectory. Equations (17)–(21) show the specific formula

$$a_n(t) = f_{\mathrm{ASL}}(\Delta v_n(t\tau : t\Delta t), \Delta x_n(t\tau : t\Delta t) v_n(t\tau : t\Delta t)), \quad (17)$$

$$v_n(t) = v_n(t\Delta t) + a_n(t) \cdot \Delta t, \quad (18)$$

$$x_n(t) = x_n(t\Delta t) + v_n(t\Delta t) \cdot \Delta t + 0.5 \cdot a_n(t) \cdot \Delta t^2, \quad (19)$$

$$\Delta x_n(t) = x_{n-1}(t) - x_n(t), \quad (20)$$

$$\Delta v_n(t) = v_{n-1}(t) - v_n(t), \quad (21)$$

where $a_n(t)$ is the acceleration of the following vehicle at time t; f_{ASL} is the mapping of the input–output relationship within the neural network; τ is the amount of sample data entered into the model, and set to 40-time steps in this work; Δt is the time step, and set to 0.1 s in this work; $v_{n-1}(t)$ is the speed of the lead vehicle at time t; $v_n(t)$ is the speed of the following vehicle at time t; $x_{n-1}(t)$ is the position of the lead vehicle at time t; $x_n(t)$ is the position of the following vehicle at time t; $\Delta x_n(t)$ is the space headway at time t; $\Delta v_n(t)$ is the relative speed at time t.

In order to verify the sensitivity of the ASL model to normal and rainy days, in this work, the cross test in used for comparing with simulation results. Firstly, use the ASL-Normal and ASL-Rain to simulate the car-following behaviors on normal and rainy days. Figures 11 and 12 show the simulated vehicle's distance, speed, and space headway. The ASL-Normal model and ASL-Rain model perform well under their respective weather conditions. The predicted trajectory basically coincides with the raw trajectory, and there is no collision in all simulations. These show that the many-to-many structure works well.

Then use the ASL-Normal model to simulate the vehicle trajectory on the rainy data-set and use the ASL-Rain model to simulate the vehicle trajectory on the normal data-set. Figures 13 and 14 show the simulated vehicles' distances, speeds and space headways. It can be seen that when using the ASL-Normal model to simulate the rainy vehicle trajectory, the simulated trajectory is more aggressive than that using the ASL-Rain model, with a larger following speed and smaller space headway. When using the ASL-Rain model to simulate the normal vehicle trajectory, the simulated trajectory is more cautious than the ASL-Normal model, with a lower following speed and larger space headway.

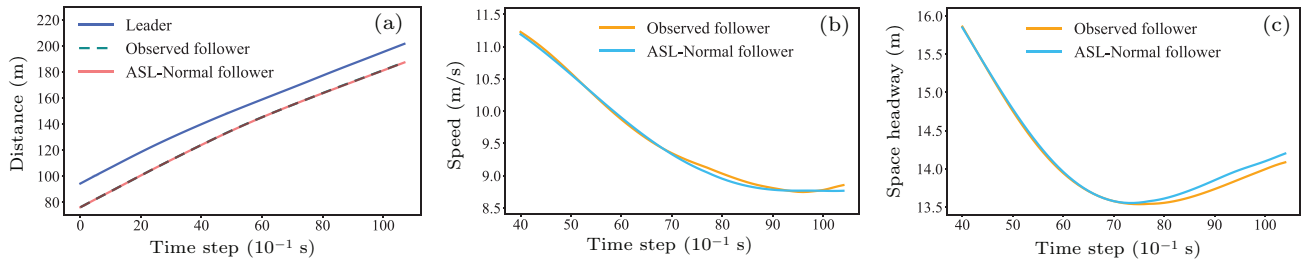


Fig. 11. ASL-Normal model simulated normal trajectory characteristics. Vehicle 389 and 4052 on normal days.

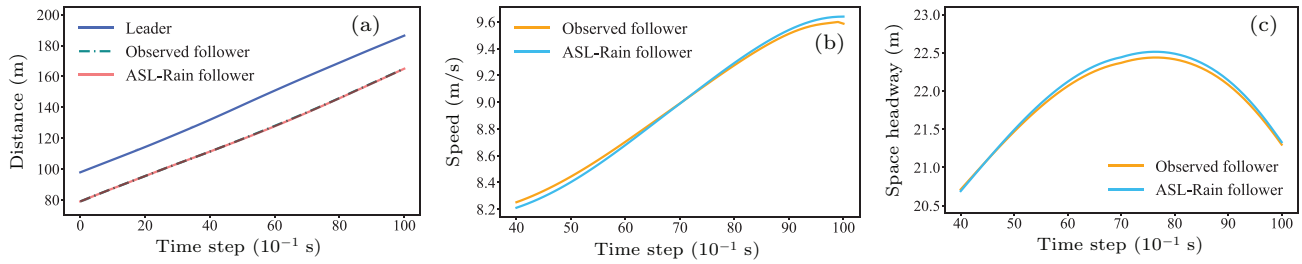


Fig. 12. ASL-Rain model simulated rainy trajectory characteristics. Vehicle 541 and 715 on rainy days.

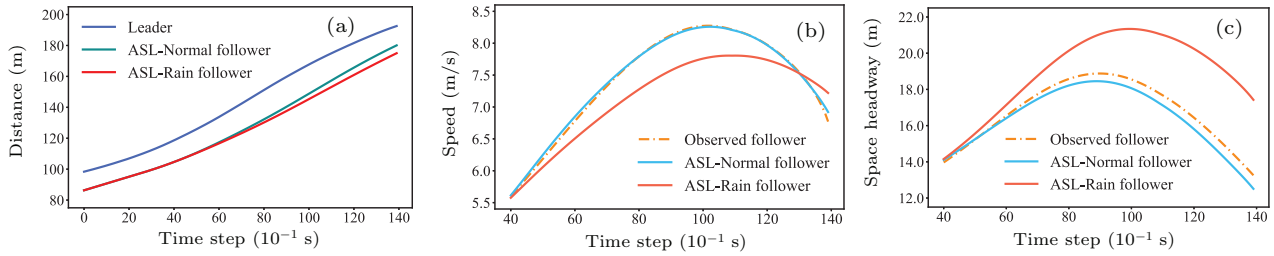


Fig. 13. ASL-Normal model and ASL-Rain model simulated normal trajectory characteristics. Vehicle 1897 and 1799 on normal day.

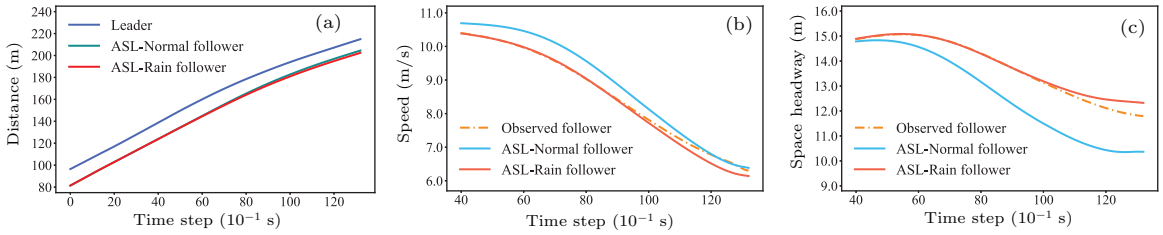


Fig. 14. ASL-Normal model and ASL-Rain model simulated rainy vehicle trajectory characteristics. Vehicle 1213 and 1450 on rainy days.

Table 5. Statistics of model cross test simulation results.

Characteristic variable	Trajectory simulation of normal data-set			Trajectory simulation of rain data-set		
	ASL-Normal	ASL-Rain	Difference	ASL-Normal	ASL-Rain	Difference
Mean speed (m/s)	12.6113	11.5431	–8.47%	11.8236	10.8784	–7.99%
Mean space headway (m)	18.0597	19.7284	9.24%	18.9825	20.6378	8.72%

Table 5 shows the statistical results of car-following space headway and speed in the cross test. Whether simulating the trajectory on normal or rainy data-sets, the following vehicle behavior simulated by the ASL-Rain model is more cautious than that by the ASL-Normal model, which indicates that the ASL car-following model is more sensitive to normal and rainy data. So, classification training or multi-scene data fusion is necessary for car-following learning on normal and rainy days.

4.2. Multi models comparison test

In order to verify the performance of the ASL model, in this work the LSTM model and the classical IDM model are selected for the simulation test. The performances of the three models in different weather scenarios are verified by comparing the two indexes of MSE and RTTC.

4.2.1. IDM model parameter calibration

As a classic car-following model, the IDM model can be applied to the description of most car-following scenes. Its advantages are that the described vehicle will not collide, the speed is stable, and the vehicle can achieve a relatively smooth transition between acceleration and deceleration.[7] The specific formulae of the IDM model are given as follows:

$$a_n(t) = a_0\left[1-\left(\frac{v_n(t)}{v_0}\right)^4-\left(\frac{\tilde{s}_n(t)}{s_n(t)}\right)^2\right], \tag{22}$$

$$\tilde{s}_n(t) = s_0 + v_n(t)T_0 + \frac{v_n(t)\cdot(v_n(t)-v_{n-1}(t))}{2\sqrt{a_0 b_0}}, \tag{23}$$

where a_0 is the following vehicle maximum acceleration, v_0 is the desired speed of the following vehicle, $\tilde{s}_n(t)$ is the desired space headway at time t, $s_n(t)$ is the space headway at time t, s_0 is the minimum space headway required for vehicles, T_0 is the desired time headway, b_0 is the comfortable deceleration.

In this work the scikit-opt code library in Python is utilized to realize the genetic algorithm,[47] which is for calibrating the parameters of the IDM model. Set the population size to 300, the maximum iteration number to 300, and the stagnation number to 100, then the minimum error will be obtained when the objective function converges to 10^{-6}.[48] For the car-following trajectory data on normal and rainy days, the calibration parameters are set to be in the range as shown in Table 6. Select root mean square percentage error (RMSPE) of space headway as the objective function[49] as shown below:

$$E_{\mathrm{RMSPE}} = \sqrt{\frac{\sum_{i=1}^{N}\left(s_i^{\mathrm{obs}}-s_i^{\mathrm{sim}}\right)^2}{\sum_{i=1}^{N}\left(s_i^{\mathrm{obs}}\right)^2}}, \tag{24}$$

where s_i^{obs} is the actual space headway s_i^{sim} is the simulation space headway. Taking the method of repeated calibration, and selecting the calibration value with the slightest error as the final result, the obtained results are shown in Table 7.

Table 6. IDM model parameter range to be calibrated.

Calibration parameters	Range
Maximum acceleration a_0 (m/s^2)	[0.1, 4.0]
Comfortable deceleration b_0 (m/s^2)	[0.1, 4.0]
Desired speed v_0 (km/h)	[1, 72]
Desired time headway T_0 (s)	[0.1, 5.0]
Congestion space headway s_0 (m)	[0.1,10]

Table 7. Optimal parameters of IDM mode.

Parameters	Normal conditions				Rainy conditions			
	Mean	SD	Max	Min	Mean	SD	Max	Min
a_0 (m/s^2)	1.4342	1.1236	3.3673	0.6876	1.3101	1.1243	2.0268	0.3098
b_0 (m/s^2)	2.5272	1.6266	3.6591	0.7128	2.7274	1.6458	4.000	0.4211
v_0 (km/h)	60.4562	4.5528	72.000	11.6863	61.7076	4.3340	72.000	14.5022
T_0 (s)	1.0706	1.0202	1.9580	0.2821	1.1588	1.2033	1.7210	0.3294
s_0 (m)	3.3616	1.9695	5.5135	0.4165	3.5819	1.9789	4.7666	0.6000

4.2.2. Discussion on model accuracy

The MSE function can express the model's prediction accuracy.[50] The smaller the value of MSE, the higher the model's prediction accuracy is. Since the MSE of speed and distance can reflect the MSE of acceleration and space headway, in this work counted are only the MSEs of speed and distance with simulated and raw trajectories in the data-set on normal and rainy days. As shown in Table 8, Figs. 15 and 16, the mean distance MSEs and the mean speed MSEs of the simulation trajectories predicted by the ASL model are the smallest under both normal and rainy conditions, which indicates that the prediction accuracy of the ASL model is the highest. Under normal and rainy conditions, the distance accuracies of the ASL model is better than those of the IDM model by 13.00% and 11.49%, and better than those of the LSTM model by 14.09% and 12.36%. The speed accuracies of the ASL model are better than those of the IDM model by 12.02% and 10.68%, and better than those of the LSTM model by 15.41% and 13.51%.

Table 8. MSE statistics results of different models.

Model	MSE of distance				MSE of speed			
	Mean	SD	Max	Min	Mean	SD	Max	Min
ASL-Normal	1.9054	6.0941	77.2653	0.0002	0.1252	0.1839	1.2405	0.0005
LSTM-Normal	2.2180	7.3795	94.5709	0.0004	0.1480	0.2285	1.9461	0.0007
IDM-Normal	2.1902	11.1232	149.4676	0.0023	0.1423	0.4289	5.3902	0.0033
ASL-Rain	2.1030	3.9993	30.0510	0.0011	0.1505	0.2042	1.4396	0.0015
LSTM-Rain	2.3996	7.2610	77.5869	0.0015	0.1740	0.2474	2.1012	0.0021
IDM-Rain	2.3759	11.2068	149.5187	0.0018	0.1685	0.4234	5.4098	0.0027

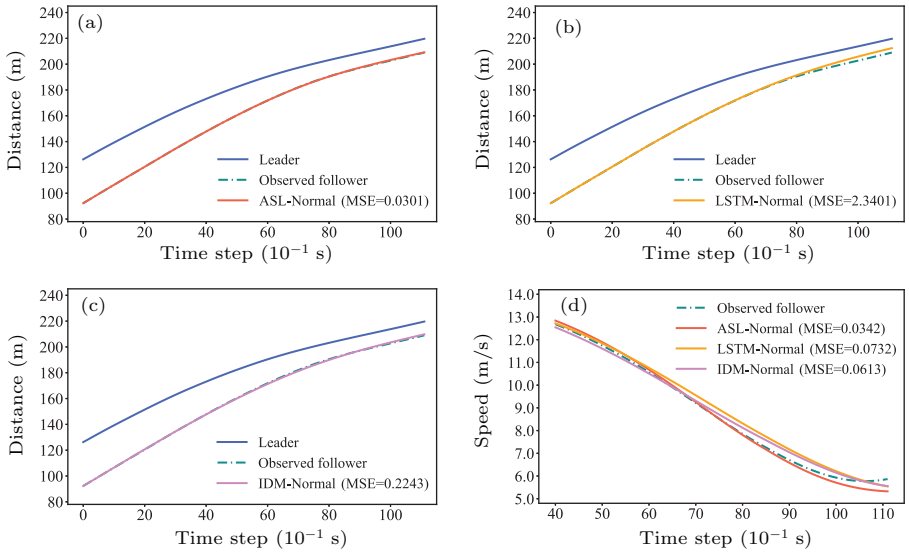


Fig. 15. Comparison of simulation trajectories among three models. Vehicle 4341 and 547 on normal days.

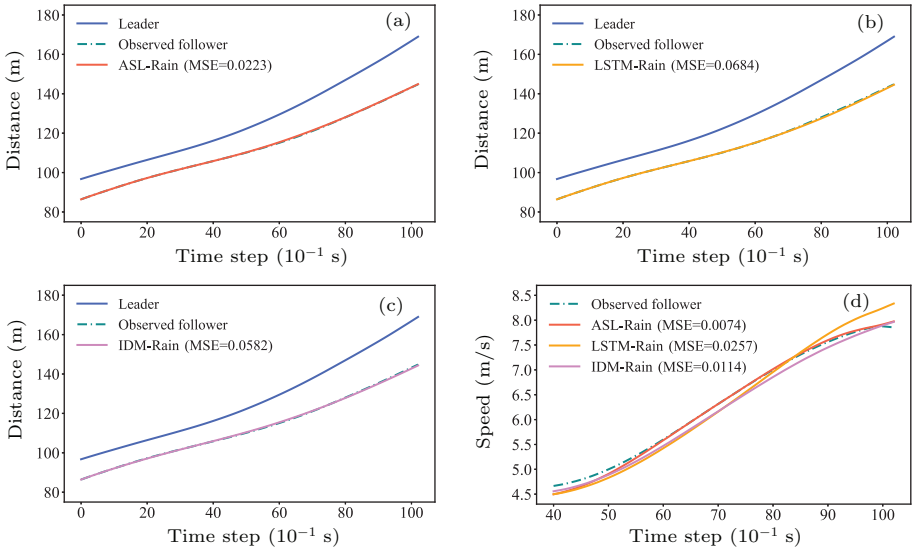


Fig. 16. Comparison of simulation trajectories among three models. Vehicle 4004 and 4044 on rainy days.

It is worth mentioning that in the IDM model the parameters of each trajectory need to be calibrated separately in order to obtain optimal accuracy. If the average value of each calibration parameter is used as the parameter value of the IDM, the MSE of the IDM will be significantly lower than that of the ASL model. The generalized neural network car-following model is better than the traditional car-following model.

The distance MSE variance and the speed MSE variance of the IDM model are greater than those of the ASL model and LSTM model. The maximum MSE values of distance and speed are also maximum values. These show that the results predicted by the IDM model are more volatile for different types of trajectories. At present, relevant studies have proved that the traditional car-following model is equivalent to the neural network car-following model in describing macro driving behavior, but it is worse than the neural network in describing some micro driving behaviors such as aggressive and cautious driving.[25,32]

The mean MSE of distance and speed of the three models under rainy conditions are greater than those under normal conditions, which shows that the precision of both the neural network model and IDM model decrease in the rainy car-following scenario. Careful driving on rainy days weakens the mutual stimulation between vehicles, and the change of acceleration does not have strong regularity, which reduces the accuracy of the three models' description of rainy data.

4.3. Discussion on driving safety

The TTC is an index widely used in the field of traffic safety to measure the safety of car-following.[51] The RTTC is the reciprocal of TTC, which can more intuitively represent the risk of collision. Specifically, the RTTC(t) is given by

$$\mathrm{RTTC}(t) = \frac{v_n(t) - v_{n-1}(t)}{x_{n-1}(t) - x_n(t) - l} \forall v_{n-1}(t) < v_n(t), \quad (25)$$

where l is the vehicle length. If $v_{n-1}(t) \geq v_n(t)$, the value of RTTC is equal to 0. The slighter the RTTC value, the higher the trajectory safety is.

Table 9 shows the RTTC statistics of the raw data and the simulation trajectories of the three models. Figure 17 shows the RTTC distribution box diagram. It can be seen from the figure that the RTTC mean of the ASL simulation trajectories on normal and rainy days are the lowest, which are 2.45%, 8.42%, and 10.47% lower than the original trajectory, IDM, and LSTM model simulation trajectories respectively on normal data, and 3.95%, 9.33%, and 11.92% on rainy data. From the standard deviation comparison of RTTC, it can be seen that the RTTC distribution of ASL model has the smallest fluctuation and is the most stable. All the above shows that the ASL model has the best security performance under normal and rainy conditions.

In addition, from the comparison between the RTTC of the raw trajectory and the simulated trajectory on normal and rainy days, we can find that the RTTC value of the trajectory on normal days is greater than on rainy days. From the perspective of speed and space headway, drivers on rainy days tend to keep a lower speed and a larger space headway for car-following, and the trajectory safety on rainy days is higher than on normal days.

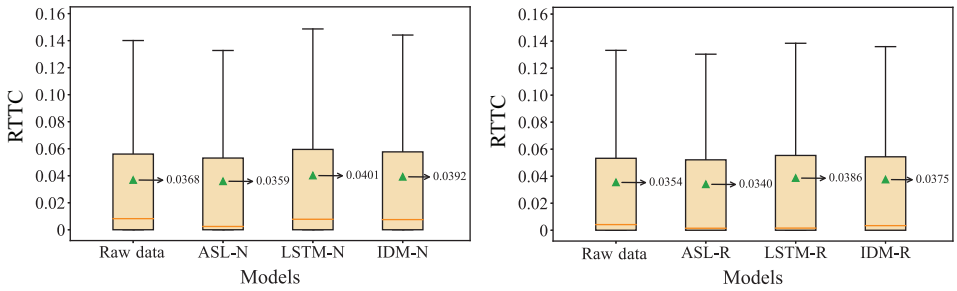


Fig. 17. RTTC distributions of different models on normal and rainy days. (a) RTTC of normal model, (b) RTTC of rainy model.

Table 9. RTTC statistic results of different models.

Model	Mean of RTTC	SD of RTTC	Model	Mean of RTTC	SD of RTTC
Raw data	0.0368	0.0591	Raw data	0.0354	0.0556
ASL-Normal	0.0359	0.0575	ASL-Rain	0.0340	0.0529
LSTM-Normal	0.0401	0.0700	LSTM-Rain	0.0386	0.0625
IDM-Normal	0.0392	0.0685	IDM-Rain	0.0375	0.0612

4.4. Traffic flow simulation

In this work the multi-vehicle linear simulation is used to verify the model's stability. The multiple vehicles are evenly distributed in one lane. The first vehicle runs according to the set state, while the subsequent cars run according to the simulated data. Combining the characteristics of the data collected in this work and referring to relevant literature,[52] the driving state of the first vehicle is set to the following four parts:

1. driving 150 s at a constant speed of 5 m/s;
2. accelerating for 10 s at 0.5 m/s^2;
3. decelerating for 10 s at 0.5 m/s^2;
4. driving 90 s at a constant speed of 5 m/s.

The platoon size is ten vehicles, seting the space headway between the following vehicles to 20 m, the initial speed to 5 m/s, and the simulation time step to 0.1 s. Figures 18 and 19 show the speed and distance of the ASL-Normal model and ASL-Rain model by multi-vehicle linear simulation, $\Delta\bar{v}_1$ represents the average deceleration amplitude of the following vehicles, $\Delta\bar{v}_2$ represents the difference in average speed between the maximum car-following speed of the following vehicle and the maximum speed of the first vehicle, and Δct represents the convergence time.

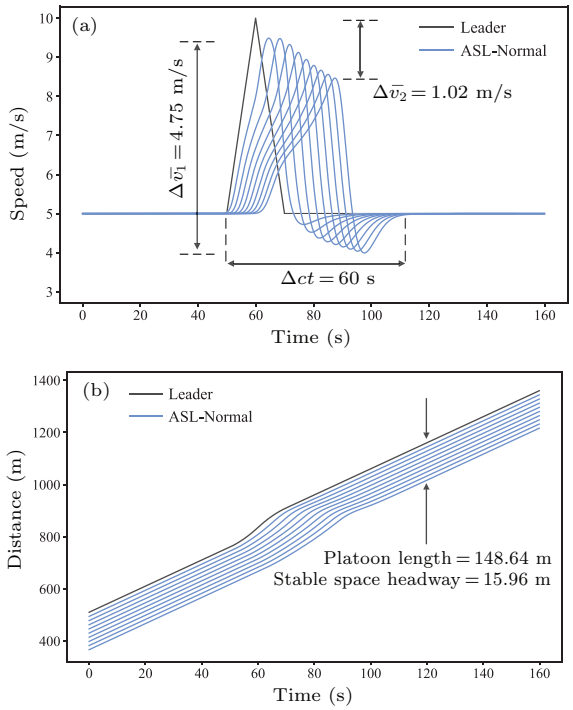


Fig. 18. Multi-vehicle straight line simulation trajectory based on ASL-Normal model. (a) Simulation speed, (b) simulation distance.

From Figs. 18(a) and 19(a), the normal and rainy car-following platoons can return to a steady state again. The stabilities of the two models are excellent. However, the car-following platoon on normal days only take 60 s from disturbance to the stable speed of 5 m/s, while 72 s on rainy days. The efficiency of the car-following platoon on normal days is high, and cautious driving on rainy days will reduce the traffic flow efficiency. Meanwhile, the car-following speed on normal days is higher than on rainy days. From the image width of Figs. 18(b) and 19(b), the car-following space headway on normal days is also smaller than on rainy days. These characteristics show that car-following behavior of the ASL-Normal model is more aggressive than the ASL-Rain model. Owing to the fast speed and small space headway on normal days, the following vehicle must adopt a large deceleration behavior to keep the vehicle from a collision when the first vehicle suddenly decelerates. As a result, the car-following platoon produces a speed of less than 5 m/s when decelerating, and the final average deceleration amplitude is 4.75 m/s. However, the rainy driver's car-following speed is slow, and the space headway is large. When the first vehicle suddenly decelerates, there is a sufficient deceleration time, and the change of car-following speed is relatively gentle. Therefore, no speed below 5 m/s occurs, and the final average deceleration amplitude is 3.74 m/s.

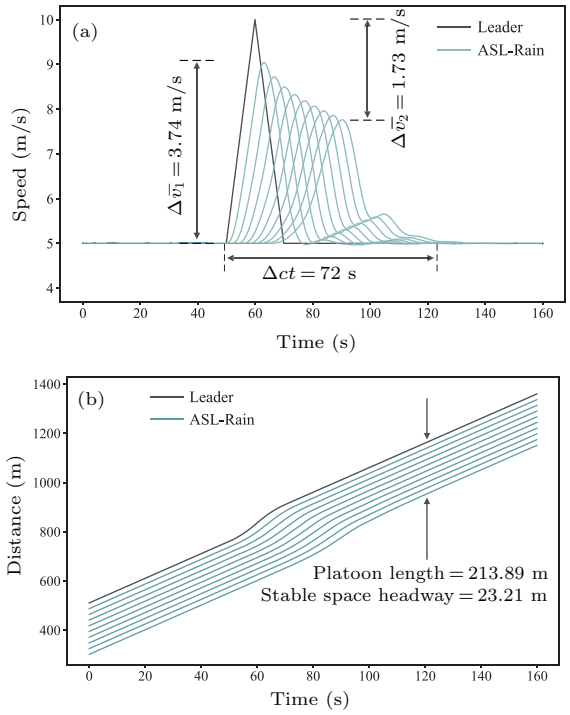


Fig. 19. Multi-vehicle straight line simulation trajectory based on ASL-Rain model. (a) Simulation speed, (b) simulation distance.

In order to further explore the differences in macro traffic flow between the ASL-Normal model and ASL-Rain model, in this paper the multi-vehicle circuit simulation is used to simulate the car-following traffic flow.[53] Initially, 20 vehicles are evenly distributed on the ring road with a perimeter of 400 m, and the space headway is 20 m. The initial speeds of all vehicles are set to 5 m/s, the time step is set to 0.1 s, and the simulation time to 2000 steps. Figures 20 and 21 show the simulation results.

Figure 20 shows the time–space map of the traffic flow without disturbance to the first vehicle. Since the initial states of all vehicles are the same, the whole system can reach a steady state soon. Both models reach the steady-state before 40 s, but the stable speed of the ASL-Normal model is 9.91 m/s, and the average speed of the system is 9.20 m/s, while the stable speed of the ASL-Rain model is only 7.97 m/s and the average speed of the system is 7.45 m/s. With the same density, the maximum traffic volume of the ASL-Normal model is 315 veh/h higher than that of the ASL-Rain model, so the efficiency of the traffic flow system on normal days is higher than on rainy days.

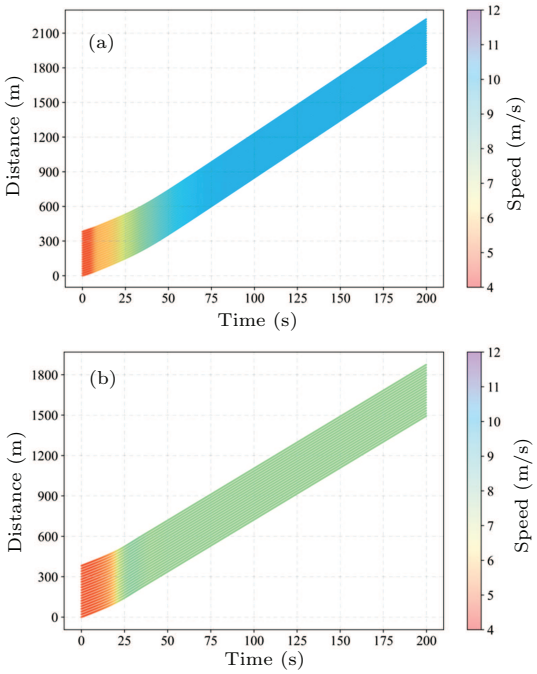


Fig. 20. Time–space map of platoon circuit simulation (without disturbance). (a) ASL-Normal model, (b) ASL-Rain model.

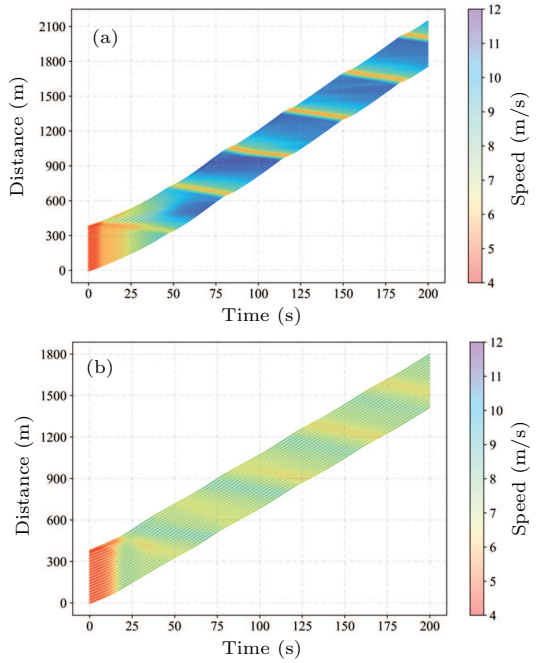


Fig. 21. Time–space map of platoon circuit simulation (disturbance of the first vehicle 5 m forward). (a) ASL-Normal model, (b) ASL-Rain model.

Figure 21 shows the time–space map of traffic flow with disturbance to the first vehicle. The position of the first vehicle moves 5 m forward so that the distance between the first vehicle and the second vehicle becomes 15 m, and the distance between the first vehicle and the last vehicle becomes 25 m. From the comparison between Figs. 21(a) and 21(b), the two traffic flow systems produce traffic oscillation and traffic congestion due to the disturbance.[54,55] However, the volatility of the traffic flow system simulated by the ASL-Normal model is more obvious than the ASL-Rain model, which produces more serious oscillation and congestion. The average system speed of the ASL-Normal model becomes 8.72 m/s, which is 5.22% lower than that under the undisturbed condition, while the average system speed of the ASL-Rain model becomes 7.27 m/s, which is only 2.42% lower than that under the undisturbed condition. The difference in maximum traffic volume between the ASL-Normal model and the ASL-Rain model decreases to 261 veh/h. All the above shows that the disturbance has a more significant influence on the sunny traffic flow system, and its traffic efficiency decreases faster.

In addition, the ASL-Normal model and ASL-Rain model can reproduce traffic oscillation's periodic propagation characteristics by adding the same disturbance. From the comparison between Figs. 21(a) and 21(b), the oscillation frequency of the traffic flow system on normal days is higher than on rainy days, but the oscillation dissipates faster. This indicates that the more aggressive driving behavior will shorten the cycle and duration of the traffic oscillation, thereby making the oscillation phenomenon more frequent.

5. Conclusions

According to the microwave radar data of urban elevated traffic flow on normal and rainy days, we train and build the ASL-Normal model and ASL-Rain model for normal and rainy days by using the LSTM optimized by AM and SSA. Through the cross test, we validate the ASL model's sensitivity to car-following data under different weather conditions. Through the simulation comparison of multiple models, we illustrate the superiority of ASL model in the performances of accuracy and safety. Through the platoon disturbance simulation, the traffic flow stability of the ASL model is verified. Overall, this study shows that the ASL model can better learn car-following behavior on normal and rainy days and can accurately describe the differences between different weather conditions. However, the model's sensitivity to normal and rainy data requires classification training to improve accuracy. Meanwhile, as far as the application of automatic driving scene is concerned, it should be further considered to establish an intelligent neural network model that can automatically select different car-following behaviors according to weather conditions.

Acknowledgements

Project supported by the National Natural Science Foundation of China (Grant No. 52072108), the Natural Science Foundation of Anhui Province, China (Grant No. 2208085ME148), and the Open Fund for State Key Laboratory of Cognitive Intelligence, China (Grant No. W2022JSKF0504).

References

[1] Hjelkrem O A and Ryeng E O 2016 *Accid. Anal. Prev.* **95** 227
[2] Chakrabarty N and Gupta K 2013 *Procedia. Soc. Behav. Sci.* **104** 1048
[3] Soria I, Elefteriadou L and Kondyli A 2014 *Simul. Modell. Pract. Theory.* **40** 208
[4] Hammit B E, Ghasemzadeh A, James R M, Ahmed M M and Young R K 2018 *Transp. Res. Part F* **59** 244
[5] Zheng L, Zhu C, He Z B, He T and Liu S S 2019 *Transportmetrica B* **7** 765787
[6] Li L, Jiang R, He Z B, Chen X Q and Zhou X S 2020 *Transp. Res. Part C* **114** 225
[7] Treiber M, Hennecke A and Helbing D 2000 *Phys. Rev. E* **62** 1805
[8] Jiang R, Wu Q S and Zhu Z J 2001 *Phys. Rev. E* **64** 017101
[9] Gazis D C, Herman R and Rothery R W 1961 *Oper. Res.* **9** 545
[10] Helbing D and Tilch B 1998 *Phys. Rev. E* **58** 133
[11] Bando M, Hasebe K, Nakayama A, Shibata A and Sugiyama Y 1995 *Phys. Rev. E* **51** 1035
[12] Gipps P G 1981 *Transp. Res. Part B* **15** 105
[13] Newell G F 1961 *Oper. Res.* **9** 209
[14] Huang Y, Yan X D, Li X M, Duan K, Gao Z J and Rakotonirainy A 2022 *Transportmetrica A*
[15] Wei D L and Liu H C 2013 *Transp. Res. Part B* **47** 1
[16] Papathanasopoulou V and Antoniou C 2015 *Transp. Res. Part C* **55** 496
[17] He Z B, Zheng L and Guan W 2015 *Transp. Res. Part B* **80** 185
[18] Yu Y, He Z B and Qu X B 2021 *IEEE Trans. Cybern.* **53** 1405
[19] Chen T Y, Wong Y D, Shi X P and Yang Y Y 2021 *Accid. Anal. Prev.* **154** 106061
[20] Li Y F, Lu X D, Ren C and Zhao H W 2019 *IEEE Access.* **7** 162778
[21] Kehtarnavaz N, Groswold N, Miller K and Lascoe P 1998 *IEEE Trans. Veh. Technol.* **47** 694
[22] Panwai S and Dia H F 2007 *IEEE Trans. Intell. Transp. Syst.* **8** 60
[23] Chong L, Abbas M M and Medina A 2011 *Transp. Res. Rec.* **2249** 44
[24] Zheng J, Suzuki K and Fujita M 2013 *Transp. Res. Part C* **36** 339
[25] Zhou M F, Qu X B and Li X P 2017 *Transp. Res. Part C* **84** 245
[26] Morton J, Wheeler T A and Kochenderfer M J 2017 *IEEE Trans. Intell. Transp. Syst.* **18** 1289
[27] Huang X L, Sun J and Sun J 2018 *Transp. Res. Part C* **95** 346
[28] Zhang X H, Sun J, Qi X and Sun J 2019 *Transp. Res. Part C* **104** 287
[29] Wang X, Jiang R, Li L, Lin Y L and Wang F Y 2019 *Physica A* **514** 786
[30] Wang X, Jiang R, Li L, Lin Y L, Zheng X H and Wang F Y 2018 *IEEE Trans. Intell. Transp. Syst.* **19** 910
[31] F. Hui, C. Wei, W. ShangGuan, R. Ando, S. Fang 2022 *Physica A* **593** 126869
[32] Ma L J and Qu S R 2020 *Transp. Res. Part C* **120** 102785
[33] Zhu M X, Wang Y H, Pu Z Y, Hu J Y, Wang X S and Ke R M 2020 *Transp. Res. Part C* **117** 102662
[34] Shi H T, Zhou Y, Wu K S, Wang X, Lin Y X, Ran B 2021 *Transp. Res. Part C* **133** 103421
[35] Hu Y P, Li Y, Huang H L, Lee J, Yuan C and Zou G Q 2022 *Accid Anal Prev.* **165** 106503
[36] Ali Y, Sharma A, Haque M M, Zheng Z and Saifuzzaman M 2020 *Accid Anal Prev.* **144** 105643
[37] Zong F, Wang M, Tang J J and Zeng M 2022 *Physica A* **589** 126625
[38] Xu J S, Hilker N, Turchet M, Al-Rijleh M K, Tu R, Wang A, Fallahshorshani M, Evans G and Hatzopoulou M 2018 *Transp. Res. Part D* **62** 90
[39] National Research Council HCM2010: *highway capacity manual*, 5th edn. 2010 Transp. Res. Board
[40] Montanino M and Punzo V 2015 *Transp. Res. Part B* **80** 82
[41] Khodayari A, Ghaffari A, Kazemi R and Braunstingl R 2012 *IEEE Trans. Syst. Man. Cybern. Part A* **42** 1440
[42] Thiemann C, Treiber M and Kesting A 2008 *Transp. Res. Rec: J. Transp. Res. Board.* **2088** 90
[43] Wang Z 2020 *Expressway Traffic Flow Characteristic Analysis and Prediction in Rainy Environment*, MS dissertation (Beijing: Beijing Jiaotong University) (in Chinese)
[44] Xu F J 2017 *Calculation and Analysis of Urban Road Capacity Under Rainfall Condition*, MS dissertation (Xi'an: Changan University) (in Chinese)
[45] Xue J K and Shen B 2020 *Syst. Sci. Control. Eng.* **8** 2234
[46] Jiang T, Sun H, Dai Y G and Liu D 2020 *J. Phys. Conf. Ser.* **1607** 012001
[47] Heinrich K, Zschech P, Janiesch C and Bonin M 2021 *Decis. Support. Syst.* **143** 113494
[48] Zhu M X, Wang X S, Tarko A and Fang S E 2018 *Transp. Res. Part C* **93** 425
[49] Sangster J, Rakha H and Du J 2013 *Transp. Res. Rec.* **2390** 20
[50] Qu X B, Wang S A and Zhang J 2015 *Transp. Res. Part B* **73** 91
[51] Vogel K 2003 *Accid. Anal. Prev.* **35** 427
[52] Jian M Y and Shi J 2020 *Saf. Sci.* **122** 104536
[53] Ding H, Pan H, Bai H J, Zheng X Y, Chen J and Zhang W H 2022 *Physica A* **596** 127154
[54] Chen D J, J. Laval, Zheng Z D and Ahn S 2012 *Transp. Res. Part B* **46** 744
[55] Laval J A 2011 *Transp. Res. Part B* **45** 385