AI Empowered Communication Systems for Intelligent Transportation Systems

Zhihan Lv, *Senior Member, IEEE*, Ranran Lou, and Amit Kumar Singh, *Senior Member, IEEE*

***Abstract*—*Intelligent* control of traffic has significant influence on the scheduling efficiency of urban traffic flow. Therefore, in order to improve the efficiency of vehicles at intersections, first, the Back Propagation (BP) neural network is used to propose a vehicle passing model at the intersection, and based on the intelligent traffic control system model, the Earliest Deadline First (EDF) dynamic scheduling algorithm is used to improve the Controller Area Network (CAN) communication network. Finally, the simulation test is used to evaluate the effectiveness of the proposed model and the improved CAN bus communication network. The results show that the neural network model can be used to predict the passage time of vehicles queuing at intersections with an error of less than 10%. The improved CAN bus communication can improve the data transmission rate, and the success rate of data transmission under different load rates is above 95%. In conclusion, the application of artificial intelligence technology in intelligent traffic system can improve the efficiency of vehicle scheduling and the efficiency of communication system. This research is of great significance to improve the communication performance of the transportation system and scheduling efficiency.**

***Index Terms*—CAN bus, BP neural network, EDF algorithm, intersection neural network model, load rate.**

I. Introduction

With the rapid development of social economy, the number of vehicles increases year by year, and the problem of traffic congestion arises [1]–[3]. Therefore, in order to improve the scheduling efficiency of the traffic control system, it is very important to realize the intelligentization of the urban traffic scheduling system. At the same time, industrial communication network, an important part of the control system, is also developing continuously [4]. However, with the rapid development of technology and network business, network performance is getting more and more attention from people. People's requirements for network speed and service content are getting higher and higher, network load is getting heavier, and network performance is greatly reduced [5], [6]. CAN was originally developed as a system for solving serial communication problems of vehicles [7]. At present, studies have shown that the application of CAN bus in automobile network has the advantages of flexibility, real-ti me performance, and reliability compared with other data communication methods [8], [9].

Artificial neural network is a mathematical model used to reflect the structure of human brain. At present, a variety of neural network models have emerged, such as neural perceptron, back propagation network, Boltzman, Hopfield network, etc. [10]–[12]. Among them, BP neural network is a back-propagation network model with good nonlinear mapping capability and high fault tolerance, which can be used to solve problems in a variety of fields [13], [14]. At present, many studies have shown that applying BP neural network algorithm to the control of traffic lights in intelligent traffic systems can achieve efficient vehicle scheduling efficiency [15]. Li *et al.* (2018) applied neural networks to the prediction of road conditions, and the results show that it can improve the operational efficiency of vehicles [16]. Pan *et al.* (2019) applied the BP neural network to the traffic prediction of urban networks, and the results showed that the prediction error was low [17]. The intelligent real-time scheduling model can obtain the current running status of the vehicle, and select the most suitable plan for the vehicle's running. The neural network performs the control of the non-linear system and has good mapping and learning functions, but the control has high requirements on the hardware of the signal controller. The effective prediction of the time when the vehicle passes the intersection is of great significance to the planning of vehicle traffic. Based on the segmentation method, Ma *et al.* (2019) predicted the driving time and the influencing factors of flow with the help of real-time traffic data, which can improve the efficiency of vehicles [18]. However, few studies have applied BP neural network model to the prediction of vehicle crossing time at intersections. EDF algorithm is one of the well-known real-time scheduling algorithms. Kalaivani and Kalaiarasi (2019) used EDF algorithm to solve the network time delay problem in network control systems. The results show that the algorithm can reduce the network delay by half [19]. Chen *et al.* (2018) applied the EDF algorithm to wireless network perception and control. The results show that the proposed framework can ensure real-time communication of the network and achieve the goal of maximizing the practicability of the system [20]. Based on this, it can be

Manuscript received May 31, 2020; revised July 20, 2020; accepted August 13, 2020. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61902203 and in part by the Key Research and Development Plan - Major Scientific and Technological Innovation Projects of Shan Dong Province under Grant 2019JZZY020101. The Associate Editor for this article was A. Jolfaei. *(Corresponding author: Zhihan Lv.)*

Zhihan Lv and Ranran Lou are with the School of Data Science and Software Engineering, Qingdao University, Qingdao 266071, China (e-mail: lvzhihan@gmail.com; qdulrr@163.com).

Amit Kumar Singh is with the Department of Computer Science and Engineering, National Institute of Technology Patna, (An Institute under MHRD, Govt. of India), Patna 800005, India (e-mail: amit_245singh@yahoo.com).

Digital Object Identifier 10.1109/TITS.2020.3017183

concluded that the EDF algorithm can solve the network delay problem, and can also improve the real-time communication performance of the network.

Based on this, based on the intelligent traffic system, first, a model for predicting the passage time of vehicles at intersections is proposed using BP neural network. Then the CAN bus communication module in the system is analyzed. Finally, to further improve the flexibility and real-time performance of CAN communication system, EDF algorithm is adopted to improve the scheduling method with fixed information priority in the system.

II. Methods

A. Relevant Technologies of Intelligent Transportation Systems

Traffic signal control system is the most important part of Intelligent transportation system [21]. The main function of the traffic signal control system is to set the timing scheme of the traffic lights in the whole area. The methods commonly used in traffic signal control are timing control, induction control, and intelligent control. Among them, intelligent control refers to the control technology that has the functions of learning, reasoning, and decision-making, and can make appropriate adjustments according to the actual situation.

In Intelligent transportation systems, vehicle detection is also required, and the methods commonly used for vehicle detection are ground induction coil detection, ultrasonic detection, and image acquisition. Among them, image acquisition and detection technology mainly includes detection box setting, target contour extraction, texture recognition, filtering processing, vehicle separation and detection, etc., but image acquisition technology is vulnerable to bad weather.

CAN bus communication system and its protocol are specially designed for vehicles. It has been widely used in vehicles. It has the characteristics of multi-master control, system flexibility, high communication rate, and so on. The longest direct communication distance of CAN node can reach 10km, and the fastest communication rate can reach 1Mbps [22].

B. Construction of Intersection Model Based on Back Propagation Neural Network

BP neural network is composed of input layer, hidden layer, and output layer [23]. If the neuron i in the input layer is Pi, the neuron j in the hidden layer is k_j, and the neuron m in the output layer is am, then the mathematical expressions of the input and output of the jth neuron in the hidden layer are as follows:

$$\begin{cases} u_j^J = \sum_{i=1}^{I} w_{ij} p_{ki} \\ v_j^J = f\left(\sum_{i=1}^{I} w_{ij} p_{ki}\right) \end{cases} \tag{1}$$

In the equation, w_{ij} is the weight of I neurons in the input layer to the j neurons in hidden layer; p_k is a network input sample; p_{ki} is the network input sample of the i-th neuron.

Then the mathematical expressions of the input and output of the mth neuron in the output layer are as follows:

$$\begin{cases} u_m^M = \sum_{j=1}^{J} w_{jm} v_j^J \\ a_{km} = f\left(\sum_{j=1}^{J} w_{jm} v_j^J\right) \end{cases} \tag{2}$$

In the equation, w_{jm} is the weight of j neurons in the hidden layer to m neurons in the output layer; v_j^J is the output value of the jth neuron in the hidden layer.

The calculation equation for the output error of the mth neuron in the output layer is as follows.

$$e_{km}(n) = d_{km}(n) - a_{km}(n) \tag{3}$$

In the above equation, e_{km} is the output error of the m-th neuron in the output layer.

The maximum descent method is used to adjust the weight w_{jm} between the hidden layer and the output layer, and w_{jm} is modified based on the learning principle of gradient descent [20].

$$\Delta w_{jm}(n) = -\eta \frac{\partial E(n)}{\partial w_{jm}(n)} = \eta \delta_m^M(n) \cdot v_j^J(n) \tag{4}$$

$$\delta_m^M(n) = -\frac{\partial E(n)}{\partial u_m^M(n)} = e_{km}(n) \cdot f'\left(u_m^M(n)\right) \tag{5}$$

In the equation, η is study step; δ is the local gradient; u_m^M is the input of the m-th neuron in the output layer.

Then, the weight between the hidden layer and the output layer can be iterated as follows.

$$w_{jm}(n+1) = \Delta w_{jm}(n) + w_{jm}(n) \tag{6}$$

The weight value between the input layer and the hidden layer can be adjusted as follows.

$$w_{ij}(n+1) = \eta \delta_j^J(n) \cdot p_{ki}(n) + w_{ij}(n) \tag{7}$$

Based on the basic structure of BP neural network, a neural network model of intersection is constructed. The modeling is based on the three-layer BP neural network structure. The numbers of nodes in the input layer and output layer are East, West, South and North in the four directions of "crossroads". The number of nodes in the hidden layer is determined according to the empirical equation.

$$l = \sqrt{n+m} + a \tag{8}$$

In the equation, l is the number of nodes in the hidden layer; n is the number of nodes in the input layer; m is the number of nodes in the output layer; a is a constant between 3 and 8.

Based on equation above, the number of nodes in the hidden layer is determined to be between 6-11. In this study, the network model is simulated by Matlab software, and the training errors and training times are compared to determine the final number of nodes in the hidden layer.

Finally, the intersection model based on BP neural network constructed in this study is shown in Fig. 1 for its specific structure.

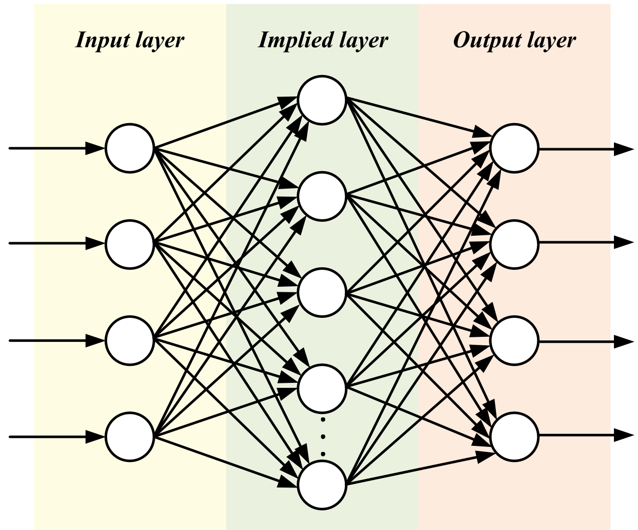

Fig. 1. Intersection model based on BP neural network.

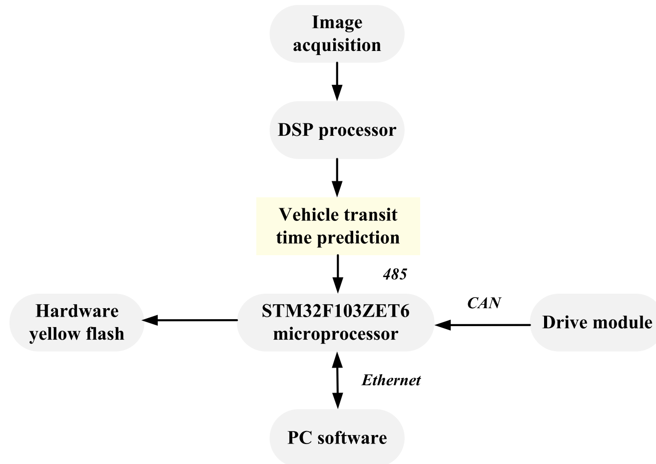

Fig. 2. The basic structure of intelligent traffic control system.

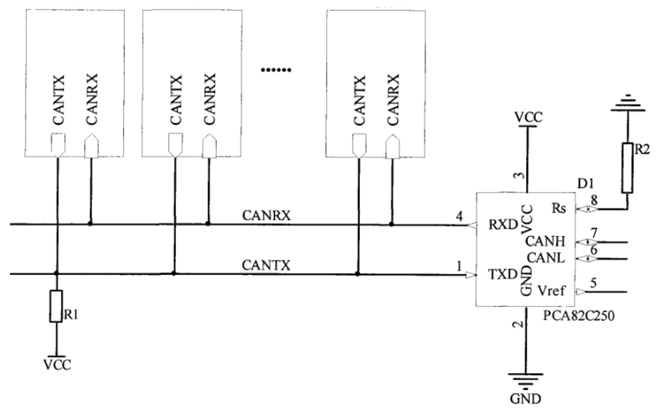

Fig. 3. Schematic diagram of CAN communication module.

C. Simulation of Intelligent Traffic Communication System Based on Can Bus

CAN bus communication system is different from other communication systems, its message transmission does not include the target address, and belongs to the whole network broadcast. The receiving station filters the data in the message according to the nature of the data, and it can realize on-line and off-line, ready-to-use, and multi-station reception, which is widely used in the field of automobile industry and so on. Then, CAN bus emphasizes the security of data, which CAN meet the requirements of data systems such as control systems. It has the following technical characteristics.

I. It follows the ISO/OSI model and applies the physical layer, data link layer, and application layer, with high communication rate, long transmission distance, and up to 110 devices.

II. Its signal transmission is short frame structure, transmission time is short, and it won't be disturbed in any way. When a node has an error, it can be closed automatically to prevent other nodes from being affected.

III. It is multi-master, any node can actively pass information to other nodes, and support point-to-point, point-to-multipoint, and global broadcast.

First, the overall structure of the intelligent traffic control system is constructed.

As shown in Fig. 2, the CAN communication system is used for the communication between the driver module (different directions of the traffic intersection) and the master control module, and each driver module has its own independent address selection pin. When the driver module receives the command sent by the master control module, it will solve the command information corresponding to its own address, and then drive the intersection to run the signal light. The schematic diagram of CAN communication module is shown in Fig. 3.

Based on this, under the simulation environment of MATLAB/Sinulink software Stateflow, the control protocol simulation model of CAN bus communication system is established and the performance is analyzed.

TABLE I

THE MOTHERBOARD MODEL MODULE IN SINULINK

Input module	Contents
Clock	The clock module, which outputs the current simulation time in each simulation
Random Number	A random number that produces a normally distributed random signal
Rounding Function	Surround rejection function, which is used to achieve the commonly used mathematical integration function;
Constant	Constant that outputs real or complex Numbers independent of time

The motherboard model module in Sinulink is shown in table I.

Among them, system Model of CAN includes 10 node modules, 1 bus module, and 1 competitive function module. All modules are parallel and the structure of each node of the node model block is the same.

The node modules include send, buffer, and period data put. Among them, period_data_put is used to collect input data. The buffer represents the node, including the null and nonull states. When the data is collected and assembled into CAN standard short frame, it will be converted from null to nonull, and the node information will be stored in the buffer. After sending, it will return to the null state. Send represents the sending part of the node. If the buffer has data waiting to be transmitted, it will be converted from sleep to wait and enter the waiting state. The bus arbitration allows the node to send, which will be converted from wait to transmission, and then transmit data. When the data transmission is finished, it will be converted to sleep by the transmission and then wait for the next data transmission.

Bus modules include idle, busy, and space. Among them, idle represents the idle state of the bus. When the bus is idle and a node requests to send information, the function competition module will be run. The compete function will be used to arbitrate the node to be sent, and idle will be converted into busy. After the node sends the data, it is converted from busy to space by the return event. After passing a space, it returns to idle and waits for the next transfer.

When the data in the node is to be transmitted, the competition function compete determines whether the node has permission to access the bus by judging the status flag bits of each node one by one.

In this simulation environment, the transmission rate of CAN bus is set to 200 KBPS, the running time is 2 s, and each frame of message data is 100 bits long. Therefore, when the CAN bus is fully loaded, 4000 frames of data will be transmitted, that is, the total length of data frames is 400 kbit. Then, the calculation equations for Throughput, Average information delay, Communication conflict rate, Network utilization, Network efficiency, and Load completion rate are as follows.

$$Throughput = \frac{thout}{T} = \frac{thout}{2} \tag{9}$$

$$Average\ information\ delay = \frac{\sum_{i=1}^{10} yti}{10} \tag{10}$$

$$Communication\ conflict\ rate = \frac{4000 \times Load\ factor - \sum_{i=1}^{10} bi}{4000 \times Load\ factor} \tag{11}$$

$$Network\ utilization = \frac{u}{T} = \frac{u}{2} \tag{12}$$

$$Network\ efficiency = \frac{thout}{u} \tag{13}$$

$$Load\ completion\ rate = \frac{\sum_{i=1}^{10} bi}{\sum_{i=1}^{10} pi} \tag{14}$$

Among them, i represents the node number, that is, 1-10; u is the total time spent in the busy state after each run is completed; bi is the number of data frames successfully sent to the bus after the i-th node is completed; pi is the number of data frames each time the i-th node sends a request.

D. Performance Improvement of Can Bus Real-Time Communication System

Real-time system refers to a processing system that can respond to an event or data within a specified time range, process it quickly, and send the result of processing to the target address in a timely manner. CAN bus communication system is a typical real -time system. In order to further improve the flexibility and real-time performance of CAN communication system, the scheduling method with fixed information priority in the system is improved.

Earliest Deadline First (EDF) algorithm is a common dynamic priority scheduling algorithm, and the allocation of priority is directly related to time [24], [25]. In the task priority assignment, EDF conducts according to the length of the task distance deadline. The shorter the distance is, the higher the priority will be, and the lower the priority will be. The task priority assignment equation is as follows.

$$d_i(t) - t \tag{15}$$

Among them, $d_i(t)$ represents the task deadline at time t, through which the task that should be scheduled at the next time can be determined.

In the EDF scheduling algorithm, the priority of the waiting task is calculated at every moment, and the next scheduling task of the system is uncertain. Establishing relationships with other tasks waiting to be scheduled makes the system more adaptable.

Compared with the improved simulation model, the priority of the node information is no longer fixed, because the priority of the node information is allocated according to the length of the deadline, so the priority of the bus communication system information changes with time.

Among them, the main improvement is the send part of the node module, which is as follows.

I. pri state and some state migration lines are added to complete the assignment of message priority in the system.

II. resend state is added, that is, a message resend mechanism is added, and resend is determined once the cut-off period is less than 0.

The improved bus arbitration section still only needs to determine whether the current node state is equal to 1. If the judgment is equal to 1, the arbitration succeeds; otherwise, the arbitration fails.

The equation for calculating the success rate of final message sending is as follows.

$$S = \frac{b_i}{p_i} \tag{16}$$

In the equation, b_i is the number of data frames successfully sent to the bus after the *ith* node completes operation; pi $_{is}$ the number of data frames each time the ith *node* sends a request.

III. Result and Discussion

A. Simulation of Intersection Model Based on BP Neural Network

As shown in Fig. 4, with the increasing number of training times, the training average error of different numbers of hidden

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

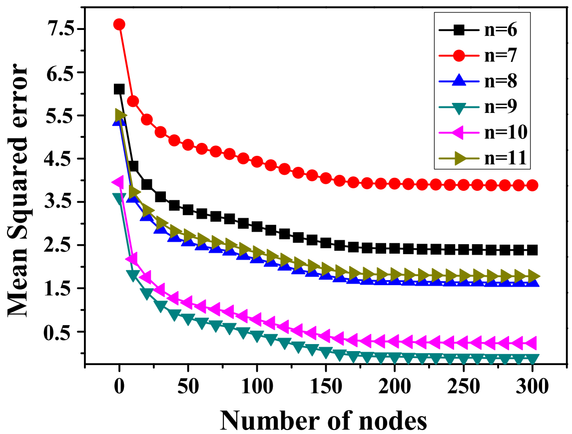

Fig. 4. Training results of intersection neural network model with different number of nodes in hidden layer (n represents the number of different nodes in hidden layer).

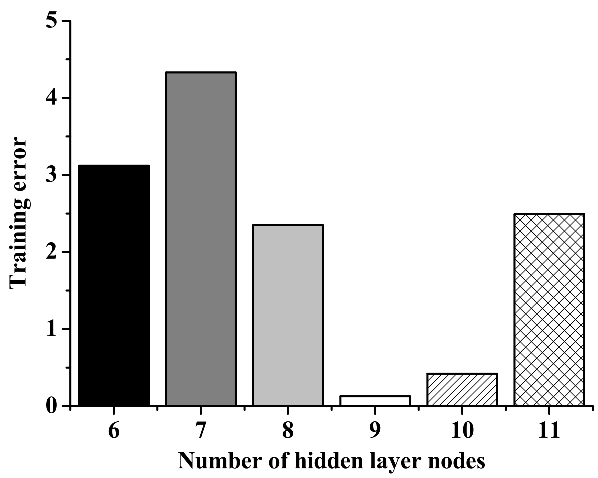

Fig. 5. Intersection neural network model training error.

layer neurons also decreases. When the number of training times exceeds 150, the training error gradually tends to level off. When the number of hidden layer nodes is 9, the MSE value is minimum.

Finally, based on the number of nodes in different hidden layers, the differences between the training errors of the intersection neural network model are compared, and the results are shown in Fig. 5. When the number of hidden layer nodes is 6, 7, 8, 9, 10 and 11, the training errors are 3.12, 4.33, 2.35, 0.13, 0.42, and 2.49, respectively.

Based on the above results, finally, the basic structure of the intersection model based on BP neural network is determined as follows. The number of nodes in the input layer is 4, the number of nodes in the hidden layer is 9, and the number of nodes in the output layer is 4. Based on this structure, the passage time of vehicles in different directions at the intersection is predicted, and the results are shown in Fig. 6. The intersection model based on BP neural network proposed in this study can effectively predict the time of vehicles passing through the intersection, which is consistent with the research results of Shen *et al.* (2012) [26]. At the same time, the prediction results of the white model for vehicles in different directions are consistent with the actual traffic time results.

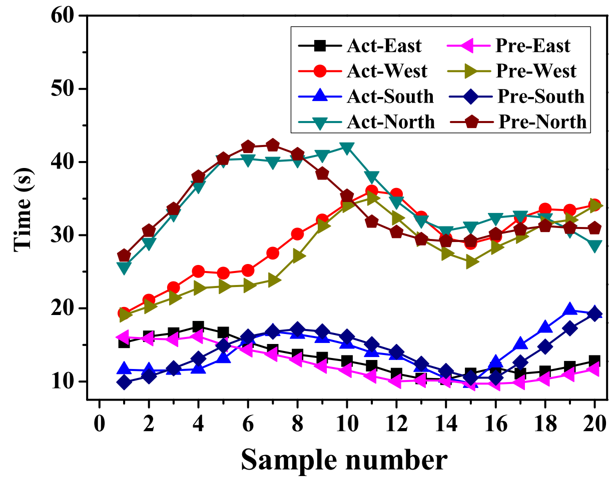

Fig. 6. The predicted output of the intersection neural network model (Act represents the actual result; Pre is the predicted result).

TABLE II
PREDICTION PERFORMANCE OF INTERSECTION NEURAL NETWORK MODEL

Direction	Number of cars	Actual passing time (s)	Predicted passing time (s)	Prediction error (%)
East	6	10.56	9.88	6.44
West	10	22.18	23.08	4.06
South	14	32.17	33.09	2.86
North	20	40.82	39.17	4.04

As shown in table II, the prediction error of the intersection model based on the BP neural network proposed in this study is less than 10% in the case of different directions and different number of vehicles.

There is a lot of randomness in the process of driving, and the driving techniques of different drivers and other factors also affect the vehicle passage time at the intersection. Therefore, there are some errors in the prediction of vehicle passage time using the model. However, the overall prediction error is less than 10%, indicating that the model constructed in this study has good prediction performance, which is consistent with the research results of Chai *et al.* (2018) [27].

B. Performance Analysis of Can Bus Communication System

The load rate is adjusted by setting the sending cycle of each node in the model, and its relationship is shown in Fig. 7. When the load rate of the network is within the range of 50%-150%, the sending cycle of different nodes is slightly different, among which the sending cycle of node 2, node 3 and node 5 is slightly higher than that of other nodes. When the load rate is in the range of 0%-50% and 150%-300%, there is no significant difference between the transmission periods of different nodes.

According to the simulation result data and (1)-(6), the influence of the load rate from 0.02 to 3.1 on network throughput, average information delay, communication conflict rate, network utilization rate, network efficiency and load

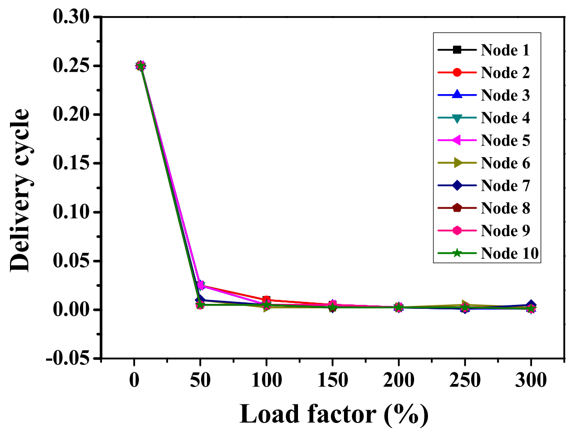

Fig. 7. The relation between delivery cycle and load rate of each node.

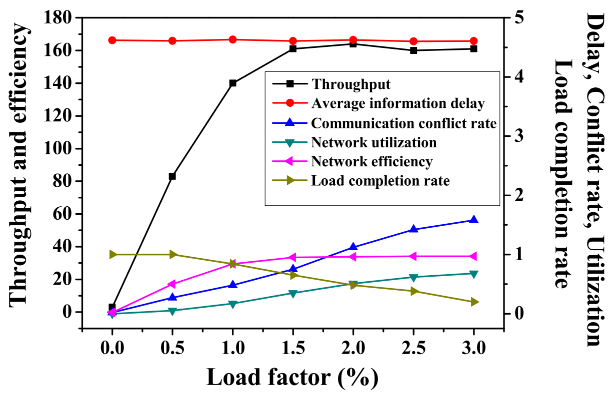

Fig. 8. The influence of load rate on the performance of communication system.

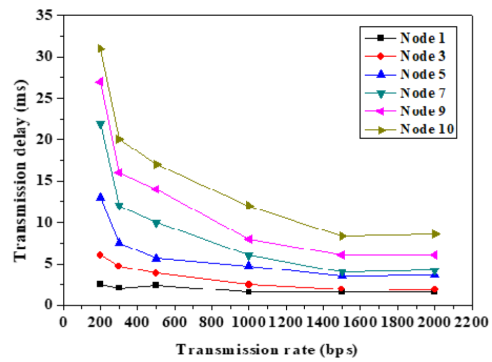

Fig. 9. The relation between transmission delay and transmission rate.

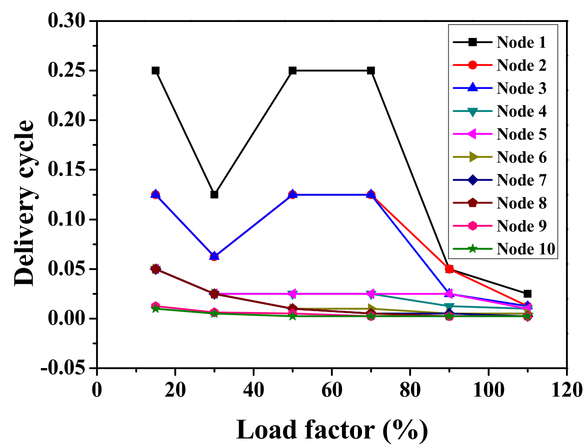

Fig. 10. The changes of sending period of each node with variation of the load rate.

completion rate is analyzed. The results are shown in Fig. 8. The change trend in the figure is determined by the CAN bus communication control protocol. When the load rate is low, low-priority information has the opportunity to compete for bus rights to be sent. With the increase of load rate, the network utilization rate will be improved, and the throughput will increase accordingly, resulting in a long time delay, and the probability of information conflicts will also increase. As a result, the network utilization rate will increase and tend to 1, while the load completion rate will decrease and be close to 1 [28], [29]. The ratio of the channel's successfully transmitted information to the channel's transmitted information per unit time hardly changes wit h the load rate. However, when the load rate increases to a certain extent, in order to avoid communication conflicts, the arbitration mechanism only sends the information with high priority, and the increment tends to be saturated [30].

When the load rate and the total running time are constant, the transmission rate changes from 200 bps to 2000 bps, and the transmission delay curve is shown in Fig. 9. When the load rate is constant, with the increase of the CAN bus transmission rate, the time for the bus to transmit information is shortened, resulting in the reduction of information transmission delay. This process is exactly the opposite of the effect of load rate increase on average information delay. The above results validate the characteristics of CAN bus communication control protocol [31], [32]. The transmission delay of node 1 and node 3 varies little with the transmission rate. When the transmission rate of node 5, node 7, node 9, and node 10 is 2000% -300%, the transmission delay decreases rapidly; when the transmission efficiency is 400%-1500%, the transmission delay decreases gradually and slowly; when the transmission efficiency is 1500%-2000%, the transmission delay is stable, which shows that the transmission delay of nodes closer to the system is less affected by the transmission rate.

C. Performance Improvement of Can Bus Real-Time Communication System

The transmission rate of CAN bus is set as 200 kbit/s by simulation, which is input by the parent template, and the total running time is 0.25s. If the data length of each packet is 100 bits, the bus transmit 500 frames of data at full load. The load rate is adjusted by setting the sending cycle of each node, and the relationship is shown in Fig. 10. The transmission periods of nodes 9 and 10 will not change with the increase of load rate. However, nodes 2 and node 3 have the same transmission cycle when the load rates are 15%, 30%, 50%, and 70%, respectively. The transmission cycle of node 1 at this load rate is roughly similar to that of node 2 and node 3, but overall higher than that of node 2 and node 3. Moreover, the

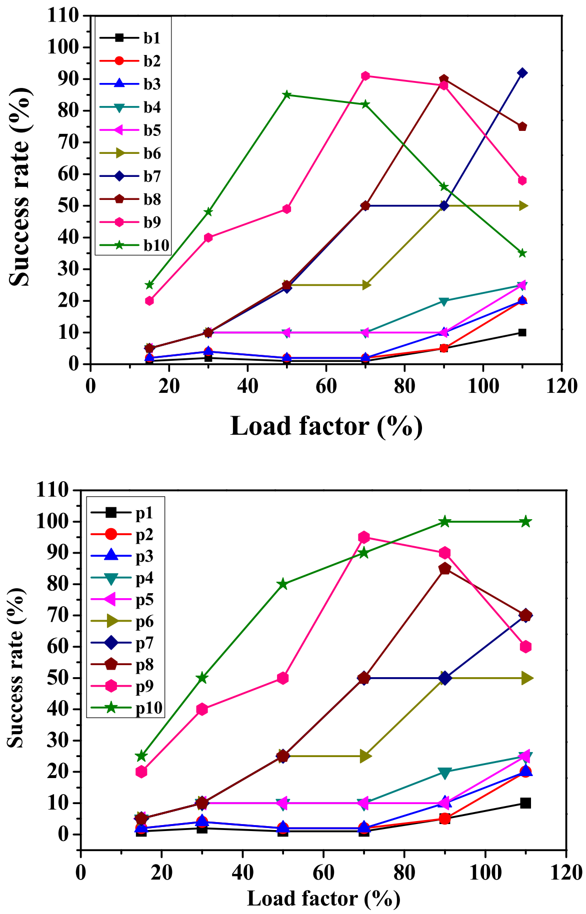

Fig. 11. The simulation results of the bus model under different load before improved. Note: The figure above shows the effect of the change in the number of data frames successfully sent before the improvement on the load rate and the success rate of CAN bus message transmission; the figure below shows the change in the number of data frames sent before the improvement on the load rate and the success rate of CAN bus message transmission influences.

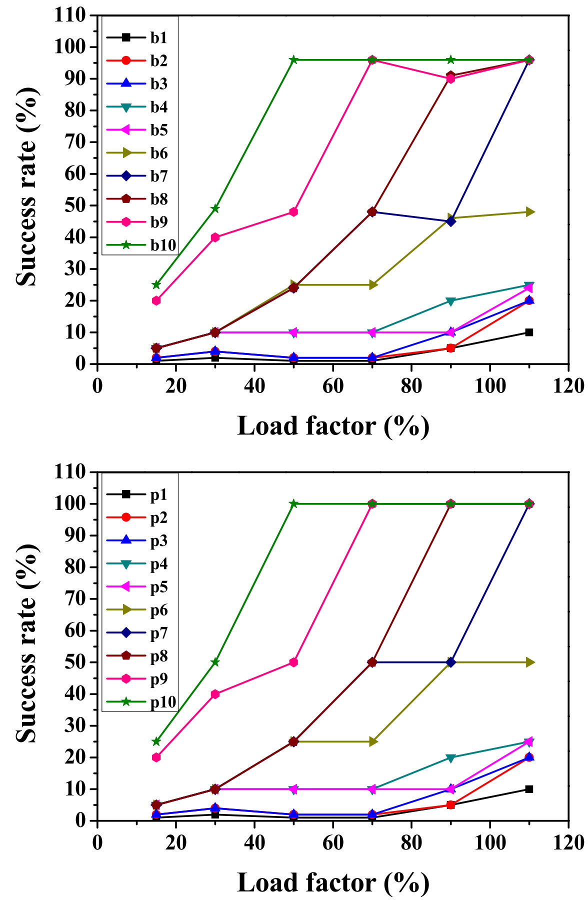

Fig. 12. Simulation results of the improved bus model under different loads. Note: The figure above shows the influence of the change of the number of frames of data successfully sent after the improvement on the load rate and the success rate of CAN bus message sending; the following figure shows the influence of the change of the frame number of the improved sending request data on the load rate and the success rate of CAN bus message sending.

overall transmission period of node 1, node 2, node 3, node 4, and node 5 shows a trend of gradual decline with the increase of load rate.

The differences between the message transmission success rate before and after the improvement were compared when the number of data frames b successfully sent to the bus by CAN bus after the completion of node operation is 1∼10, and when the number of data frames p is 1∼10 for each request sent by the node. First, the effect of increased load rate on the number of data frames b successfully sent to the bus after the completion of node operation before and after the improvement is compared. As shown in Fig. 11 and Fig. 12, when b=1, b=2, b=3, b=4, b=5, b=6, and b=7, there is no significant difference in the message transmission success rate of CAN bus before and after the improvement. When b=8, b=9, and b=10, the message transmission success rate of the improved CAN bus tends to be stable with the gradual increase of the load rate.

Then, the effect of the increase of load rate on the number of data frames p of each request sent by the node before and after the improvement is compared. As shown in Fig. 11(bottom) and Fig. 12 (bottom), when p=1, p=2, p=3, p=4, p=5, and p=6, there is no significant difference in the message transmission success rate of CAN bus before and after the improvement. When p=7, p=8, p=9, p=10, the message transmission success rate of the improved CAN bus tends to be stable with the gradual increase of the load rate.

The static priority-driven scheduling (SPDS) algorithm [8], [33] and the EDF algorithm proposed in this study are compared for the data delivery success rate when the load rate of CAN bus communication system is 50%, 70%, 90%, and 110%. The results are shown in Fig. 13. When the load rate of the system is 50%, the success rates of the CAN bus simulation model based on SPDS and the CAN bus simulation model based on EDF are both more than 80 %. As the load rate of the system increases, the nodes in the CAN bus simulation model based on SPDS decrease with the decrease of priority, so does the success rate of data transmission. However, in the CAN bus simulation model based on EDF, The data transmission rate of different nodes is significantly increased, and the data transmission success rate under different load rates is more than 95%. It shows that the improvement of the CAN bus communication system using the EDF algorithm can improve the data transmission

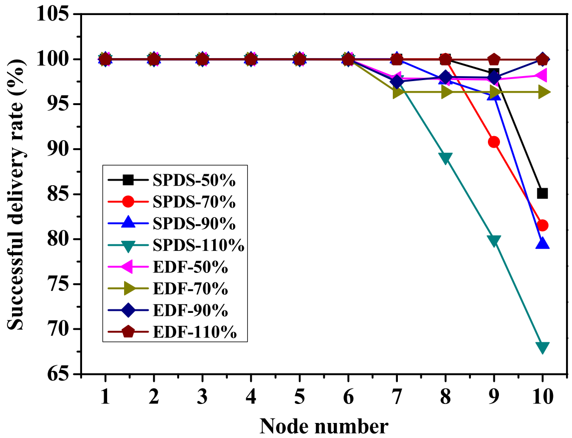

Fig. 13. Comparison of the data transmission success rate of CAN bus before and after the improvement under different load rates.

efficiency of the communication system, which is consistent with the results of Como *et al.* (2020) [34].

IV. Conclusion

In order to study the application of artificial intelligence in intelligent traffic system, a prediction model of vehicle crossing time is proposed based on BP neural network. Then, based on the communication technology in the intelligent transportation system, the simulation model is established according to the technical characteristics of CAN bus communication, and the EDF algorithm is used to improve the system. The results show that the predicted errors of the neural network model applied to the crossing time of vehicles in different directions are all less than 10%, and the data success rate of different nodes in the CAN bus simulation model based on EDF increases significantly, and the data success rate under different load rates is more than 95%. However, BP neural network has the problem of slow convergence speed and local minimum value. In the the The following research, other models will be selected for performance evaluation. In conclusion, the results of this study can provide a theoretical basis for improving the communication efficiency of intelligent transportation systems and the effect of vehicle scheduling. In future, this research can improve the other researches in 6G, IoT, AI [35]–[48].

References

[1] R. Ewing, G. Tian, and T. Lyons, "Does compact development increase or reduce traffic congestion?" *Cities*, vol. 72, pp. 94–101, Feb. 2018.

[2] S. Nakamura, T. Uemura, G. Koutaki, and K. Uchimura, "Speeding up of the traffic congestion mitigation by stochastic optimization in deep learning," *Nonlinear Theory Appl.*, vol. 9, no. 1, pp. 49–59, 2018.

[3] Y. Yao, Y. Hong, D. Wu, Y. Zhang, and Q. Guan, "Estimating the effects of 'community opening' policy on alleviating traffic congestion in large Chinese cities by integrating ant colony optimization and complex network analyses," *Comput., Environ. Urban Syst.*, vol. 70, pp. 163–174, Jul. 2018.

[4] K. Kolomvatsos, "An intelligent, uncertainty driven management scheme for software updates in pervasive IoT applications," *Future Gener. Comput. Syst.*, vol. 83, pp. 116–131, Jun. 2018.

[5] C. Sexton, Q. Bodinier, A. Farhang, N. Marchetti, F. Bader, and L. A. DaSilva, "Enabling asynchronous machine-type D2D communication using multiple waveforms in 5G," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1307–1322, Apr. 2018.

[6] I. Demirkol, D. Camps-Mur, J. Paradells, M. Combalia, W. Popoola, and H. Haas, "Powering the Internet of Things through light communication," *IEEE Commun. Mag.*, vol. 57, no. 6, pp. 107–113, Jun. 2019.

[7] S. Tariq, S. Lee, H. K. Kim, and S. S. Woo, "CAN-ADF: The controller area network attack detection framework," *Comput. Secur.*, vol. 94, Jul. 2020, Art. no. 101857.

[8] B. Groza and P.-S. Murvay, "Security solutions for the controller area network: Bringing authentication to in-vehicle networks," *IEEE Veh. Technol. Mag.*, vol. 13, no. 1, pp. 40–47, Mar. 2018.

[9] B. Groza and P.-S. Murvay, "Efficient intrusion detection with Bloom filtering in controller area networks," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 4, pp. 1037–1051, Apr. 2019.

[10] M. Yasir, S. Afzal, K. Latif, G. M. Chaudhary, N. Y. Malik, F. Shahzad, and O. Y. Song, "An efficient deep learning based model to predict interest rate using Twitter sentiment," *Sustainability*, vol. 12, no. 4, pp. 1–16, 2020.

[11] W. Lu *et al.*, "Intelligence in tourist destinations management: Improved attention-based gated recurrent unit model for accurate tourist flow forecasting," *Sustainability*, vol. 12, no. 4, p. 1390, Feb. 2020.

[12] M. A. M. Ramírez, E. Benetos, and J. D. Reiss, "Deep learning for black-box modeling of audio effects," *Appl. Sci.*, vol. 10, no. 2, p. 638, Jan. 2020.

[13] P.-C. Shih, C.-Y. Chiu, and C.-H. Chou, "Using dynamic adjusting NGHS-ANN for predicting the recidivism rate of commuted prisoners," *Mathematics*, vol. 7, no. 12, p. 1187, Dec. 2019.

[14] S. Chen, N. Wu, J. Xiao, T. Li, and Z. Lu, "Expulsion identification in resistance spot welding by electrode force sensing based on wavelet decomposition with multi-indexes and BP neural networks," *Appl. Sci.*, vol. 9, no. 19, p. 4028, Sep. 2019.

[15] X. Liang, X. Du, G. Wang, and Z. Han, "A deep reinforcement learning network for traffic light cycle control," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1243–1253, Feb. 2019.

[16] J. Li *et al.*, "An end-to-end load balancer based on deep learning for vehicular network traffic control," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 953–966, Feb. 2019.

[17] X. Pan, W. Zhou, Y. Lu, and N. Sun, "Prediction of network traffic of smart cities based on DE-BP neural network," *IEEE Access*, vol. 7, pp. 55807–55816, 2019.

[18] J. Ma, J. Chan, G. Ristanoski, S. Rajasegarar, and C. Leckie, "Bus travel time prediction with real-time traffic information," *Transp. Res. C, Emerg. Technol.*, vol. 105, pp. 536–549, Aug. 2019.

[19] C. T. Kalaivani and N. Kalaiarasi, "Earliest deadline first scheduling technique for different networks in network control system," *Neural Comput. Appl.*, vol. 31, no. 1, pp. 223–232, Jan. 2019.

[20] Y. Chen, H. Zhang, N. Fisher, L. Y. Wang, and G. Yin, "Probabilistic per-packet real-time guarantees for wireless networked sensing and control," *IEEE Trans. Ind. Informat.*, vol. 14, no. 5, pp. 2133–2145, May 2018.

[21] V. Sharma, I. You, G. Pau, M. Collotta, J. Lim, and J. Kim, "LoRaWAN-based energy-efficient surveillance by drones for intelligent transportation systems," *Energies*, vol. 11, no. 3, p. 573, Mar. 2018.

[22] H. Zhang, Y. Shi, J. Wang, and H. Chen, "A new delay-compensation scheme for networked control systems in controller area networks," *IEEE Trans. Ind. Electron.*, vol. 65, no. 9, pp. 7239–7247, Sep. 2018.

[23] X. H. Han, X. Xiong, and F. Duan, "A new method for image segmentation based on BP neural network and gravitational search algorithm enhanced by cat chaotic mapping," *Appl. Intell.*, vol. 43, no. 4, pp. 855–873, 2015.

[24] P. Bartlett, D. Helmbold, and P. Long, "Gradient descent with identity initialization efficiently learns positive definite linear transformations by deep residual networks," *Neural Comput.*, vol. 31, no. 3, pp. 477–502, 2019.

[25] E. Ghadaksaz and S. Safari, "Storage capacity for EDF–ASAP algorithm in energy-harvesting systems with periodic implicit deadline hard real-time tasks," *J. Syst. Archit.*, vol. 89, pp. 10–17, Sep. 2018.

[26] D. Cheng, X. Zhou, Y. Xu, L. Liu, and C. Jiang, "Deadline-aware MapReduce job scheduling with dynamic resource availability," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 4, pp. 814–826, Apr. 2019.

[27] X. Y. Shen, H. X. Liu, and J. Liu, "Prediction for percentage of vehicle entering expressway rest area based on bp neural network," *Appl. Mech. Mater.*, vol. 178, pp. 1956–1960, May 2012.

[28] H. Chai, X. Chen, Y. Cai, and J. Zhao, "Artificial neural network modeling for predicting wood moisture content in high frequency vacuum drying process," *Forests*, vol. 10, no. 1, p. 16, Dec. 2018.

[29] J. Zhang *et al.*, "A deep learning framework for driving behavior identification on in-vehicle CAN-BUS sensor data," *Sensors*, vol. 19, no. 6, p. 1356, Mar. 2019.

[30] H. Guo and W. Hu, "Design of distributed multi-parameter measurement control system based on CAN bus," in *Proc. IOP Conf. Ser., Mater. Sci. Eng.*, vol. 452, 2018, pp. 1–8.

[31] W. Prodanov, M. Valle, and R. Buzas, "A controller area network bus transceiver behavioral model for network design and simulation," *IEEE Trans. Ind. Electron.*, vol. 56, no. 9, pp. 3762–3771, Sep. 2009.

[32] E. Choi, S. Han, J. Lee, S. Lee, S. Kang, and J.-W. Choi, "Compatibility analysis of the turbo controller area network (TURBO CAN)," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5146–5157, Jun. 2018.

[33] J. Song, K.-E. Han, D.-U. Kim, C. Park, and K. Kim, "Priority-based grant-aware scheduling for low-latency switching," *Photon. Netw. Commun.*, vol. 36, no. 2, pp. 175–186, Oct. 2018.

[34] Y. Jiang, H. Abouee-Mehrizi, and Y. Diao, "Data-driven analytics to support scheduling of multi-priority multi-class patients with wait time targets," *Eur. J. Oper. Res.*, vol. 281, no. 3, pp. 597–611, Mar. 2020.

[35] G. Como, P. Frasca, and S. Yuksel, "Technical committee on networks and communication systems [technical activities]," *IEEE Control Syst. Mag.*, vol. 40, no. 2, pp. 19–22, Apr. 2020.

[36] Z. Lv and H. Song, "Mobile Internet of Things under data physical fusion technology," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4616–4624, May 2020.

[37] R. Shi, Y. Gan, and Y. Wang, "Evaluating scalability bottlenecks by workload extrapolation," in *Proc. IEEE 26th Int. Symp. Modeling, Anal., Simulation Comput. Telecommun. Syst. (MASCOTS)*, Sep. 2018, pp. 333–347.

[38] Z. Lv and W. Xiu, "Interaction of edge-cloud computing based on SDN and NFV for next generation IoT," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5706–5712, Jul. 2020.

[39] G. Casolla, S. Cuomo, V. S. D. Cola, and F. Piccialli, "Exploring unsupervised learning techniques for the Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2621–2628, Apr. 2020.

[40] F. Piccialli, S. Cuomo, V. S. D. Cola, and G. Casolla, "A machine learning approach for IoT cultural data," *J. Ambient Intell. Humanized Comput.*, pp. 1–12, Sep. 2019.

[41] Z. Lv, X. Li, H. Lv, and W. Xiu, "BIM big data storage in WebVRGIS," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2566–2573, Apr. 2020.

[42] S. Dang, O. Amin, B. Shihada, and M.-S. Alouini, "What should 6G be?" *Nature Electron.*, vol. 3, no. 1, pp. 20–29, Jan. 2020.

[43] F. Piccialli, G. Casolla, S. Cuomo, F. Giampaolo, and V. S. D. Cola, "Decision making in IoT environment through unsupervised learning," *IEEE Intell. Syst.*, vol. 35, no. 1, pp. 27–35, Jan. 2020.

[44] Z. Lv, W. Kong, X. Zhang, D. Jiang, H. Lv, and X. Lu, "Intelligent security planning for regional distributed energy Internet," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3540–3547, May 2020.

[45] A. Zhou, S. Wang, S. Wan, and L. Qi, "LMM: Latency-aware microservice mashup in mobile edge computing environment," *Neural Comput. Appl.*, pp. 1–15, Jan. 2020.

[46] S. Wan, Z. Gu, and Q. Ni, "Cognitive computing and wireless communications on the edge for healthcare service robots," *Comput. Commun.*, vol. 149, pp. 99–106, Jan. 2020.

[47] Z. Lv, B. Hu, and H. Lv, "Infrastructure monitoring and operation for smart cities based on IoT system," *IEEE Trans. Ind. Informat.*, vol. 16, no. 3, pp. 1957–1962, Mar. 2020.

[48] S. Wan, Y. Xia, L. Qi, Y. H. Yang, M. Atiquzzaman. "Automated colorization of a grayscale image with seed points propagation," *IEEE Trans. Multimedia*, vol. 22, no. 7, pp. 1756–1768, Jul. 2020.

[49] S. Yang *et al.*, "Scalable digital neuromorphic architecture for large-scale biophysically meaningful neural network with multi-compartment neurons," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 1, pp. 148–162, Jan. 2020.

Zhihan Lv (Senior Member, IEEE) received the Ph.D. degree from the Ocean University of China and Paris7 University, in 2012. He is currently an Associate Professor with Qingdao University, China. He has been an Assistant Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, from 2012 to 2016. He worked in CNRS, France, as a Research Engineer, Umea University, Sweden, as a Postdoctoral Research Fellow, Fundacion FIVAN, Spain, as an Experienced Researcher, University College London, U.K., as a Research Associate, and University of Barcelona, Spain, as a Postdoc. He was a Marie Curie Fellow in European Union's Seventh Framework Program LANPERCEPT. He has contributed over 200 articles in the related fields on journals such as the IEEE Transactions on Industrial Informatics, the IEEE Transactions on Intelligent Transportation, the IEEE Transactions on Fuzzy Systems, the IEEE Transactions on Systems, Man, and Cybernetics, the IEEE Transactions on Emerging Topics in Computing, the IEEE Journal on Selected Areas in Communications, the IEEE Journal of Selected Topics in Signal Processing, the IEEE Internet of Things Journal, ACM TOMM, ACM TOIT, and ACM TIST, and conferences such as ACM MM, ACM CHI, ACM Siggraph Asia, ICCV, and IEEE Virtual Reality. His research mainly focuses on the Internet of Things, blockchain, multimedia, augmented reality, virtual reality, computer vision, 3D visualization and graphics, serious game, HCI, big data, and GIS.

Dr. Lv is an Associate Editor of *Plos one* (Since 2016), IEEE Access (Since 2016), *Neurocomputing* (2016–2018), and *IET Image Processing* (Since 2017). He is the leading Guest Editor of the IEEE Transactions on Industrial Informatics, the IEEE Transactions on Intelligent Transportation Systems, the IEEE Network, the IEEE Sensors, the *IEEE Consumer Electronics Magazine, Future Generation Computer Systems, Neurocomputing*, and *Neural Computing and Applications*. He is a Program Committee member of ACM IUI2015, 2016, 2019, 2020, IEEE CHASE Workshop on BIGDATA4HEALTH 2016, 2017, IEEE/CIC WIN Workshop 2016, IIKI2016-2019, WASA2016, 2017, IEEE PDGC2016, ACM SAC2017-WCN Track, IEEE CTS2016 Workshop on IoT2016, IEEE DASC2017, 2020, ISAPE2017, IoTBDS2017, IEEE AIMS2017, IEEE iThings-2017, IEEE VTC2017-Fall, IEEE INFOCOM 2020 workshop, and ACM MobiCom 2020 workshop.

Ranran Lou received the bachelor's degree in engineering from Qingdao University in 2019, where he is currently pursuing the master's degree in software engineering with the School of Data Science and Software Engineering. He has extensive experience in marine data processing. His research directions are deep learning and big data.

Amit Kumar Singh (Senior Member, IEEE) received the Ph.D. degree from the National Institute of Technology Kurukshetra, Haryana, India, in 2015. He is currently an Assistant Professor with the Computer Science and Engineering Department, National Institute of Technology Patna, India. He has authored over 100 peer-reviewed journals, conference publications, and book chapters. He has authored three books and edited four books with internationally recognized publishers such as Springer and Elsevier. His research interests include data hiding, image processing, biometrics, and cryptography. He is the Associate Editor of the IEEE Access (Since 2016), *IET Image Processing* (Since 2020), and a former member of the editorial board of *Multimedia Tools and Applications*, Springer (2015–2019). He has edited various international journal special issues as a Lead Guest Editor. He has obtained the memberships from several international academic organizations such as ACM and IEEE.